.103 Cl. 6,8

US006347156B1

(12) **United States Patent**
Kamada et al.

(10) **Patent No.: US 6,347,156 B1**
(45) **Date of Patent: Feb. 12, 2002**

(54) **DEVICE, METHOD AND STORAGE MEDIUM FOR RECOGNIZING A DOCUMENT IMAGE**

(75) Inventors: **Hiroshi Kamada**; **Katsuhito Fujimoto**, both of Kanagawa (JP)

(73) Assignee: **Fujitsu Limited**, Kawasaki (JP)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/216,712**

(22) Filed: **Dec. 21, 1998**

(30) **Foreign Application Priority Data**

May 27, 1998 (JP) 10-145322

(51) **Int. Cl.**[7] **G06K 9/00**; G06K 9/32; G06K 9/46; H04N 1/40; H04N 1/41

(52) **U.S. Cl.** **382/237**; 382/162; 382/164; 382/165; 382/169; 382/181; 382/190; 382/299; 382/300; 358/429; 358/455

(58) **Field of Search** 382/162, 164, 382/165, 166, 169, 173, 176, 181, 190, 195, 237, 299, 321, 205, 232, 300; 358/448, 455, 500, 429

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,128,748 A * 7/1992 Murakami et al. 358/500
5,274,471 A * 12/1993 Kim 358/455
5,583,659 A * 12/1996 Lee et al. 358/455
5,742,703 A * 4/1998 Lin et al. 382/176
5,781,658 A * 7/1998 O'Gorman 382/172
5,809,167 A * 9/1998 Al-Hussein 382/190
5,850,466 A * 12/1998 Schott 382/141
5,875,268 A * 2/1999 Miyake 382/276
6,055,336 A * 4/2000 Niki 382/237

FOREIGN PATENT DOCUMENTS

JP 1-243668 9/1989
JP 2-9268 1/1990

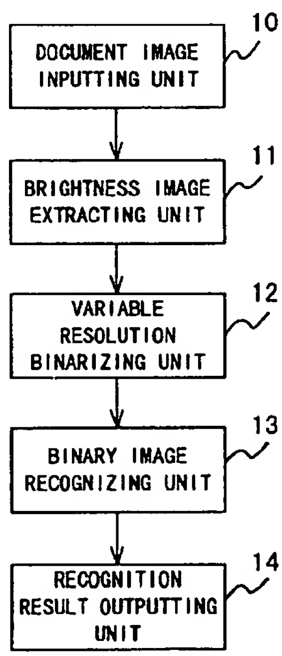
* cited by examiner

Primary Examiner—Leo Boudreau
Assistant Examiner—Daniel G. Mariam
(74) *Attorney, Agent, or Firm*—Staas & Halsey LLP

(57) **ABSTRACT**

A color image input from a document image inputting unit is converted into a gray-scale image by a brightness image extracting unit. The gray-scale image is then converted into an image having a higher resolution according to the resolution of the original gray-scale image. When this conversion is performed, subpixels are generated between the original pixels, and the values of the subpixels are obtained with an interpolation method. Furthermore, a threshold value for a binarization process is generated by using an original pixel value and a subpixel value. The characters included in the binarized image are recognized by a binary image recognizing unit, and a recognition result is output from a recognition result outputting unit.

54 Claims, 19 Drawing Sheets



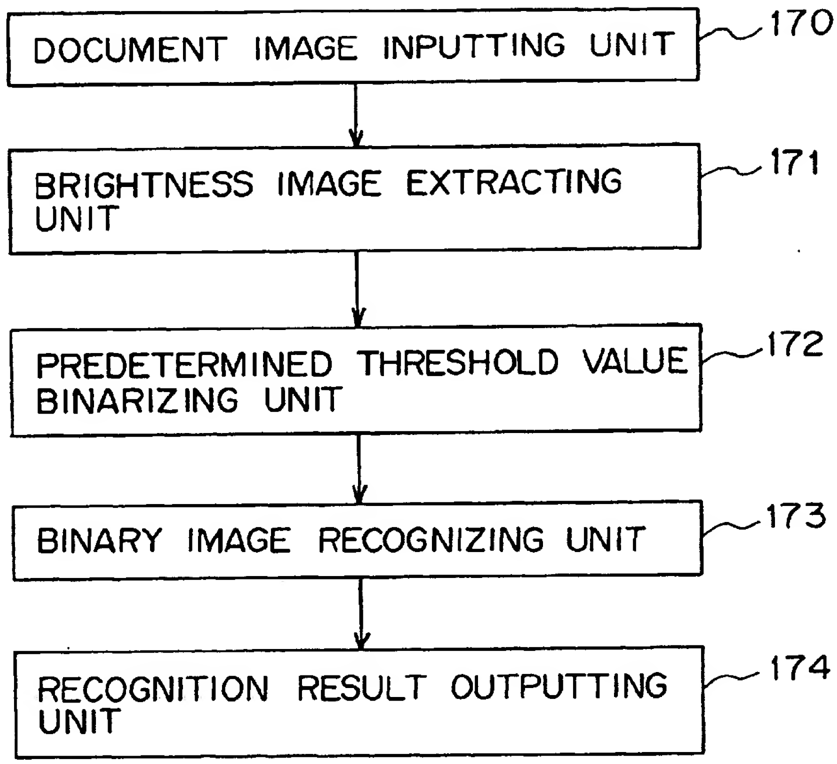
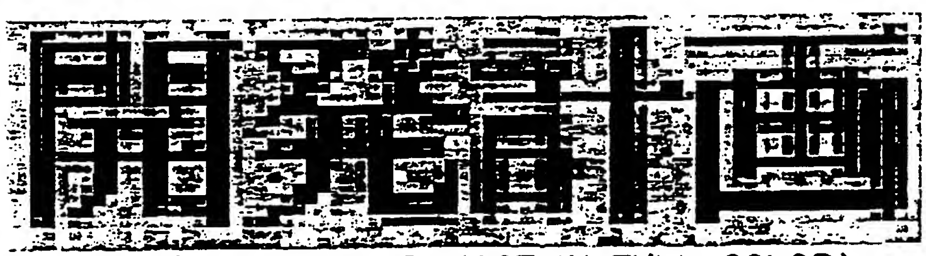


FIG. 1A PRIOR ART



(150 dpi TEXT IMAGE IN FULL COLOR)

NOTE) VARIOUS COLORS CAN BE IDENTIFIED AROUND CHARACTERS IN COLOR DISPLAY

FIG. 1B PRIOR ART

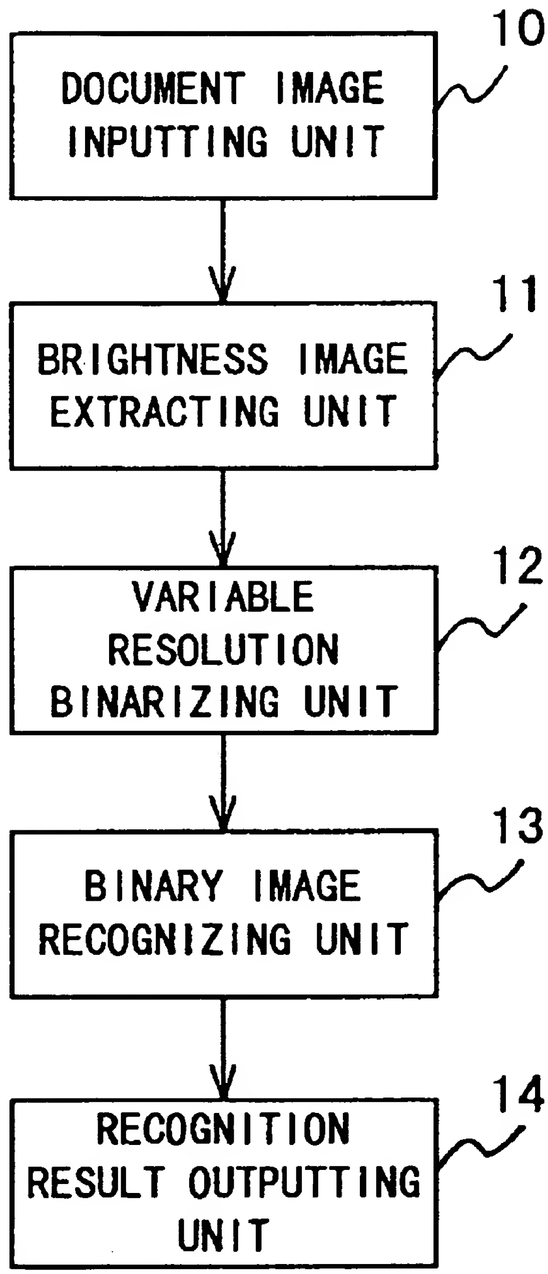

DOCUMENT IMAGE INPUTTING UNIT
10
BRIGHTNESS IMAGE EXTRACTING UNIT
11
VARIABLE RESOLUTION BINARIZING UNIT
12
BINARY IMAGE RECOGNIZING UNIT
13
RECOGNITION RESULT OUTPUTTING UNIT
14

FIG. 2

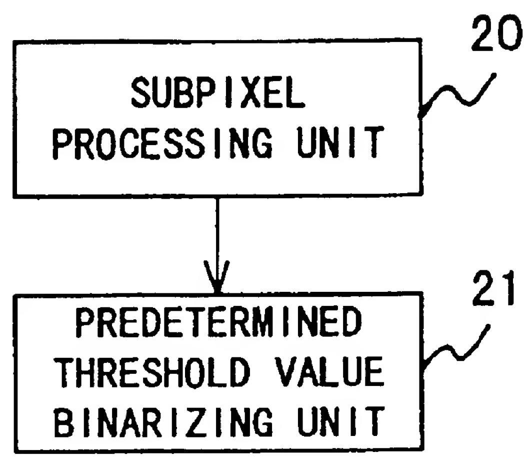

20
SUBPIXEL PROCESSING UNIT
21
PREDETERMINED THRESHOLD VALUE BINARIZING UNIT

FIG. 3A

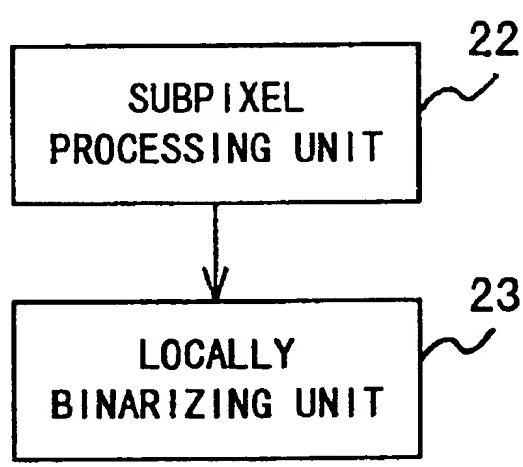

22
SUBPIXEL PROCESSING UNIT
23
LOCALLY BINARIZING UNIT

FIG. 3B

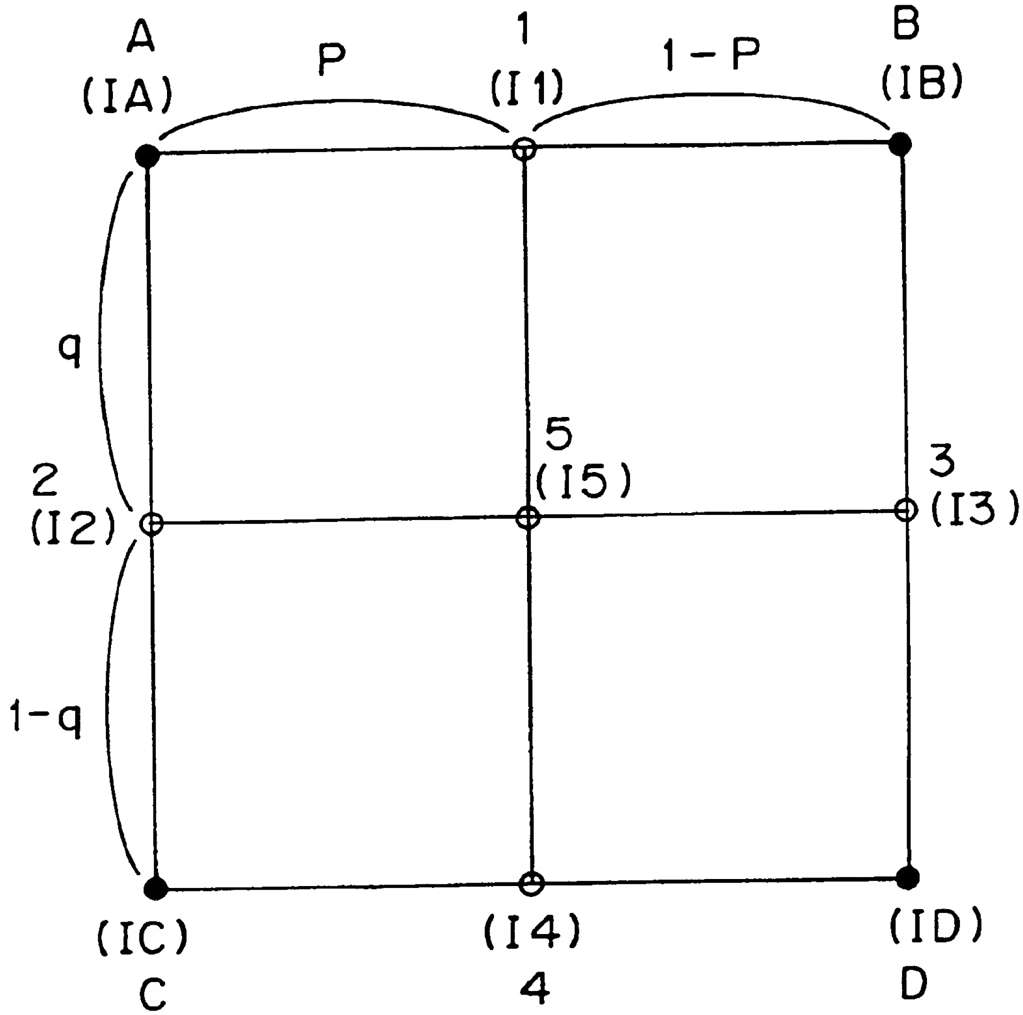


FIG. 4

RESOLUTION OF INPUT IMAGE	PARAMETER FOR SUBPIXEL GENERATION PROCESS
100	6
150	3
⋮	⋮

FIG. 5

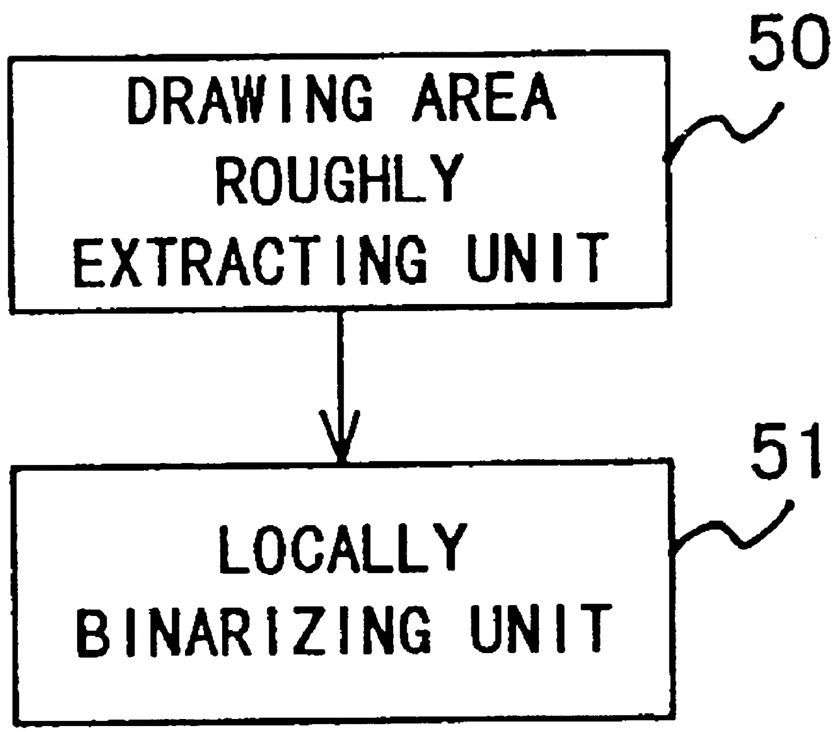

DRAWING AREA ROUGHLY EXTRACTING UNIT
50
LOCALLY BINARIZING UNIT
51

FIG. 6

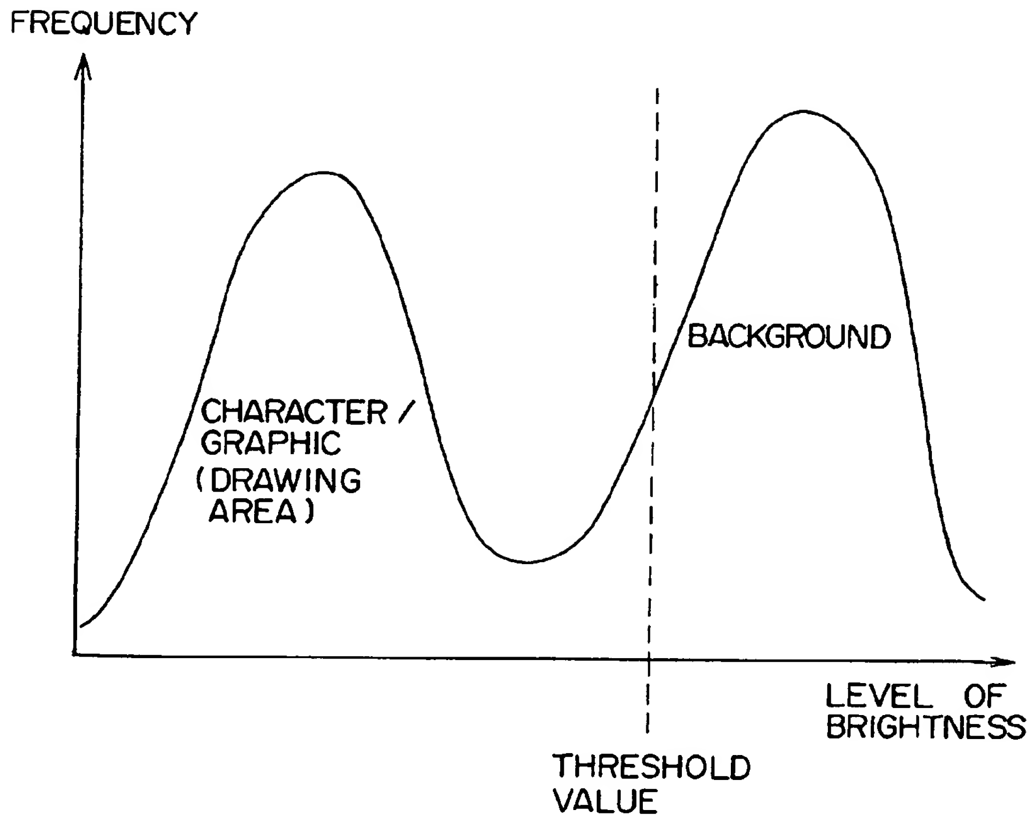

FREQUENCY
BACKGROUND
CHARACTER / GRAPHIC (DRAWING AREA)
LEVEL OF BRIGHTNESS
THRESHOLD VALUE

FIG. 7

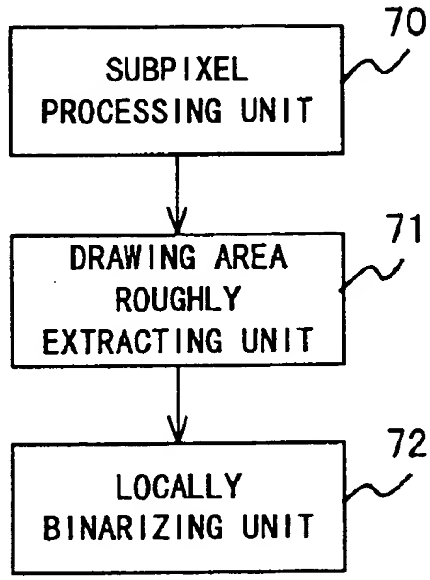

SUBPIXEL PROCESSING UNIT
70
DRAWING AREA ROUGHLY EXTRACTING UNIT
71
LOCALLY BINARIZING UNIT
72

FIG. 8A

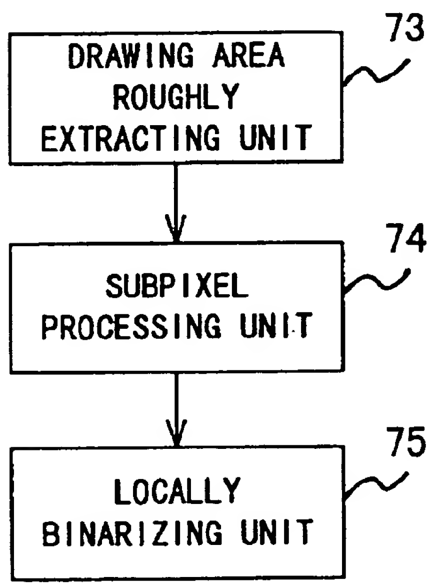

DRAWING AREA ROUGHLY EXTRACTING UNIT
73
SUBPIXEL PROCESSING UNIT
74
LOCALLY BINARIZING UNIT
75

FIG. 8B

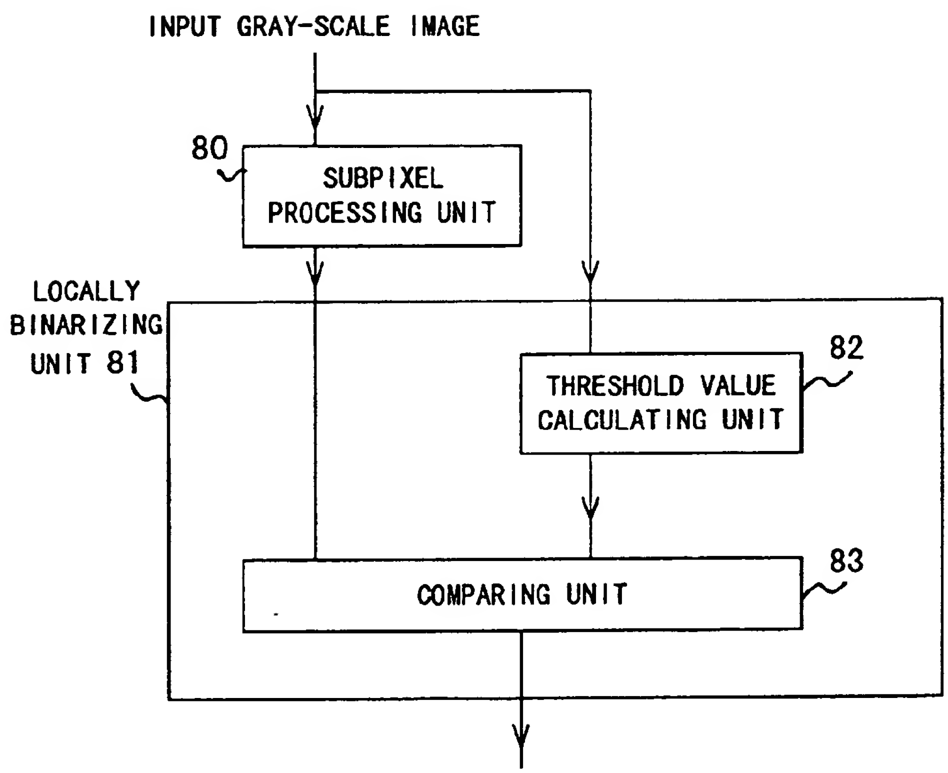

INPUT GRAY-SCALE IMAGE
80
SUBPIXEL PROCESSING UNIT
LOCALLY BINARIZING UNIT 81
82
THRESHOLD VALUE CALCULATING UNIT
83
COMPARING UNIT

FIG. 9

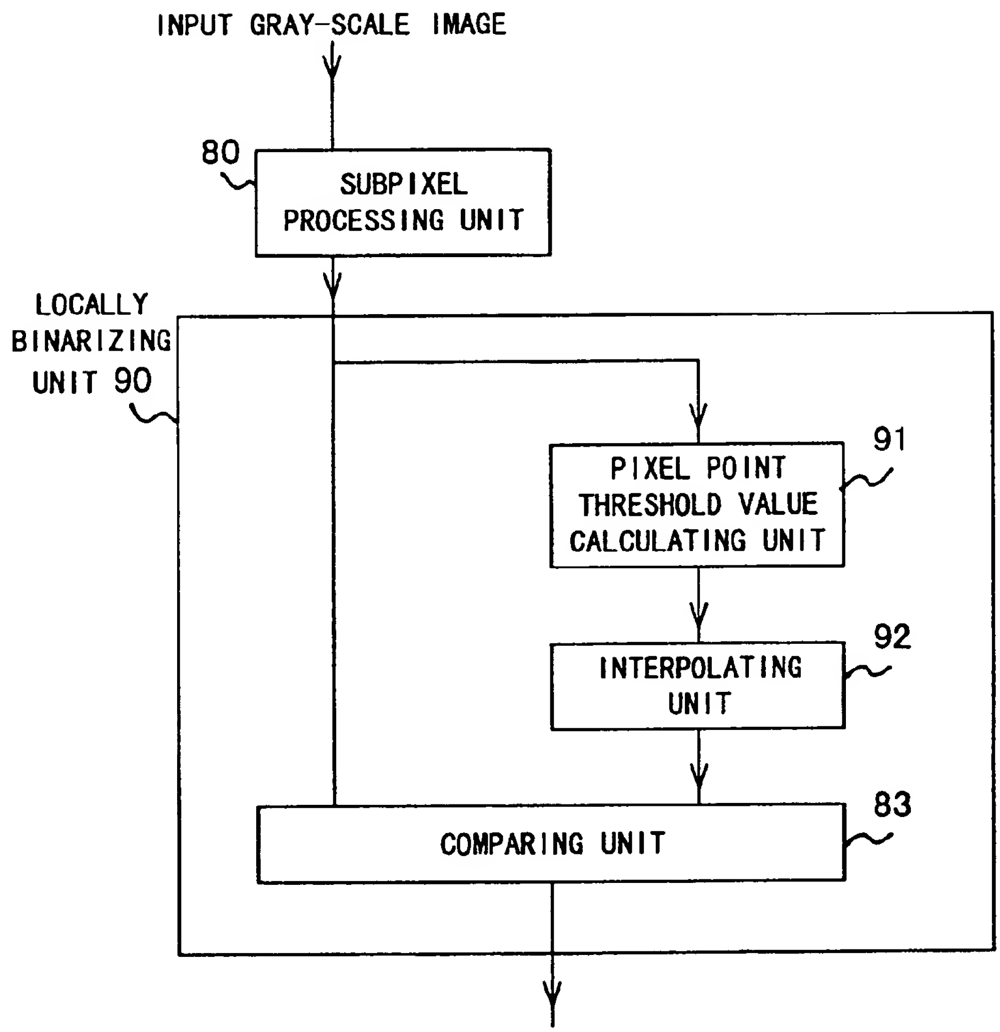

INPUT GRAY-SCALE IMAGE
80
SUBPIXEL PROCESSING UNIT
LOCALLY BINARIZING UNIT 90
91
PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
92
INTERPOLATING UNIT
83
COMPARING UNIT

FIG. 10

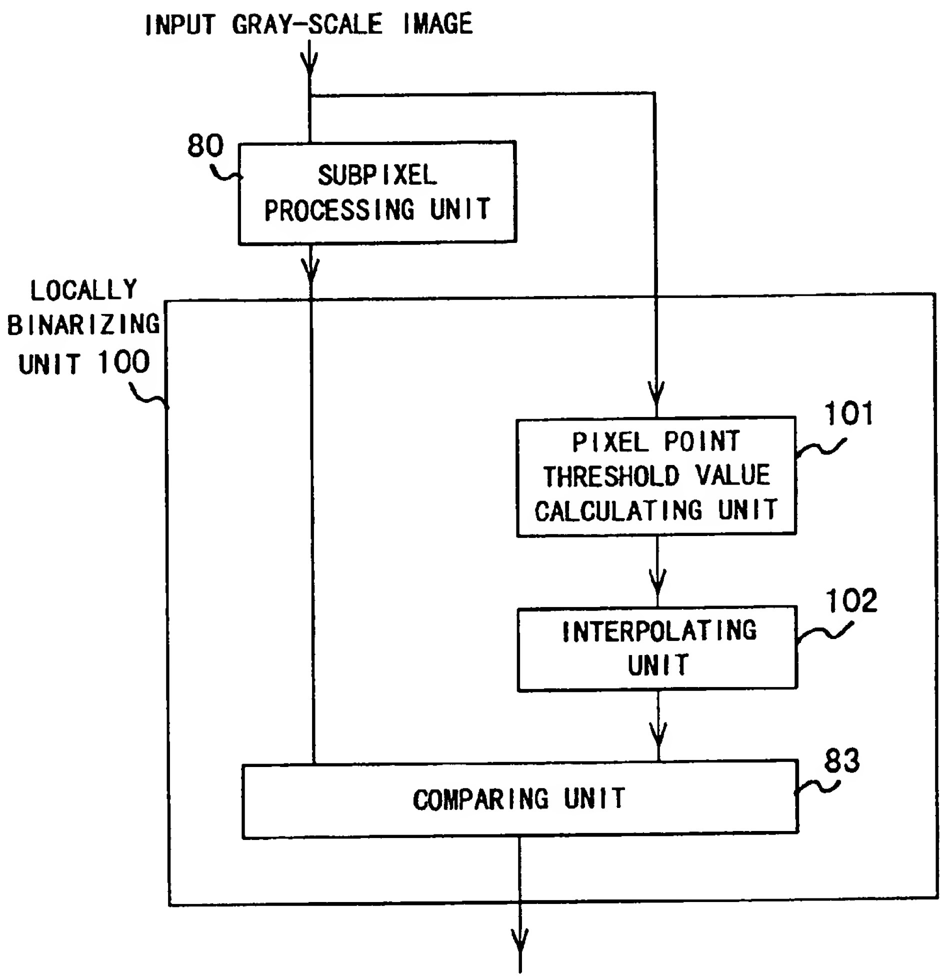

INPUT GRAY-SCALE IMAGE
80
SUBPIXEL PROCESSING UNIT
LOCALLY BINARIZING UNIT 100
PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
101
INTERPOLATING UNIT
102
COMPARING UNIT
83

FIG. 11

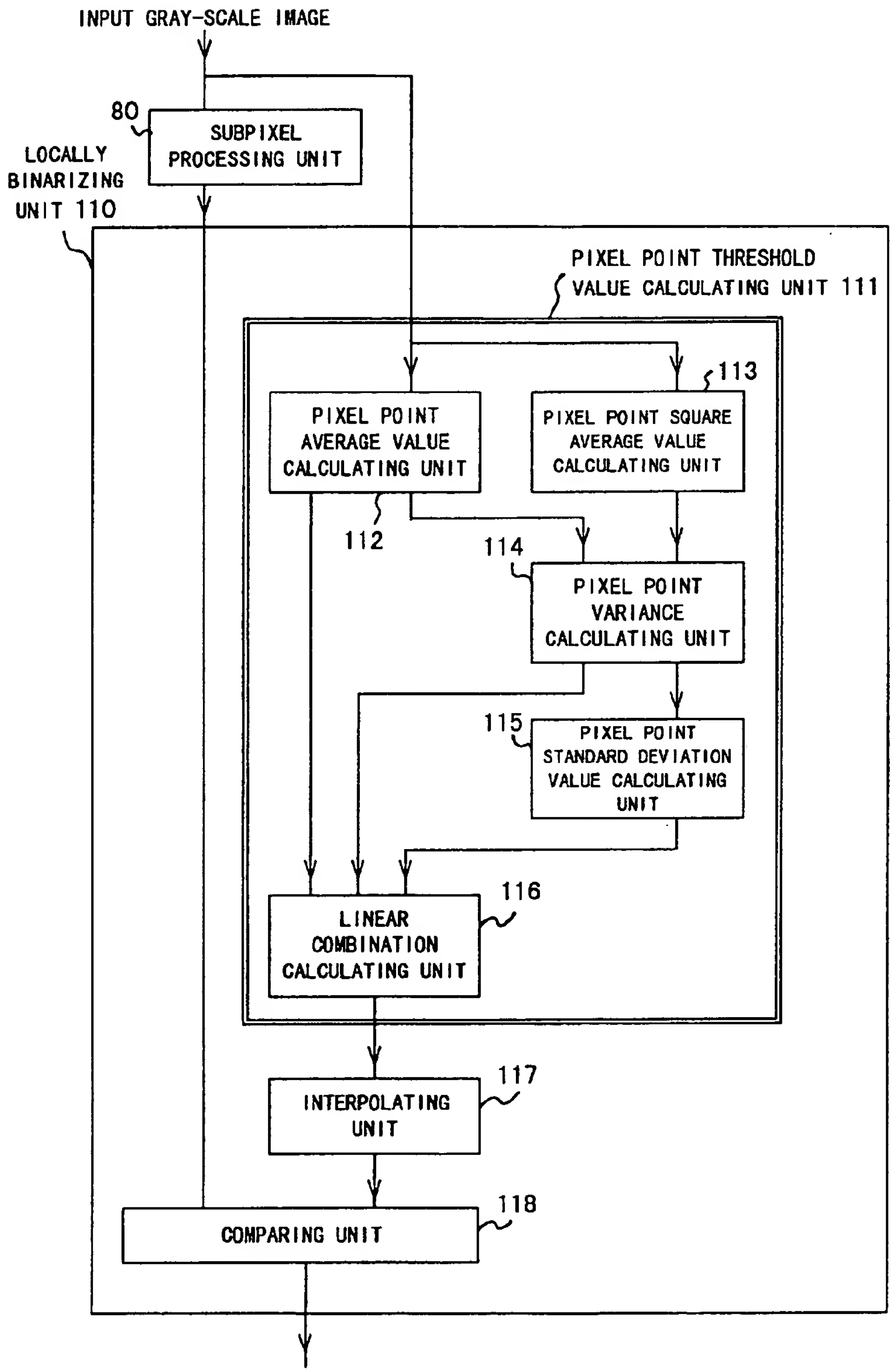

INPUT GRAY-SCALE IMAGE
80
SUBPIXEL PROCESSING UNIT
LOCALLY BINARIZING UNIT 110
PIXEL POINT THRESHOLD VALUE CALCULATING UNIT 111
113
PIXEL POINT AVERAGE VALUE CALCULATING UNIT
PIXEL POINT SQUARE AVERAGE VALUE CALCULATING UNIT
112
114
PIXEL POINT VARIANCE CALCULATING UNIT
115
PIXEL POINT STANDARD DEVIATION VALUE CALCULATING UNIT
116
LINEAR COMBINATION CALCULATING UNIT
117
INTERPOLATING UNIT
118
COMPARING UNIT

FIG. 12

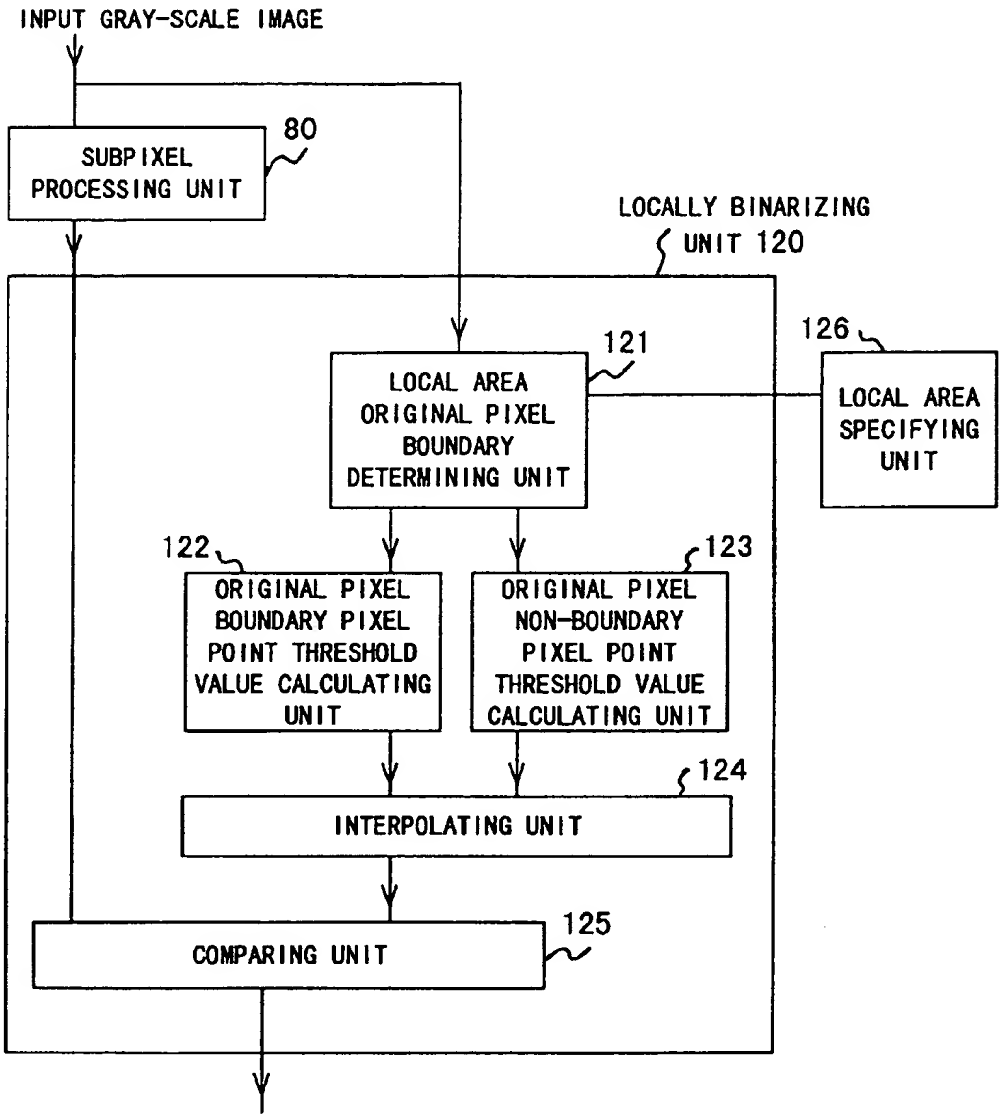
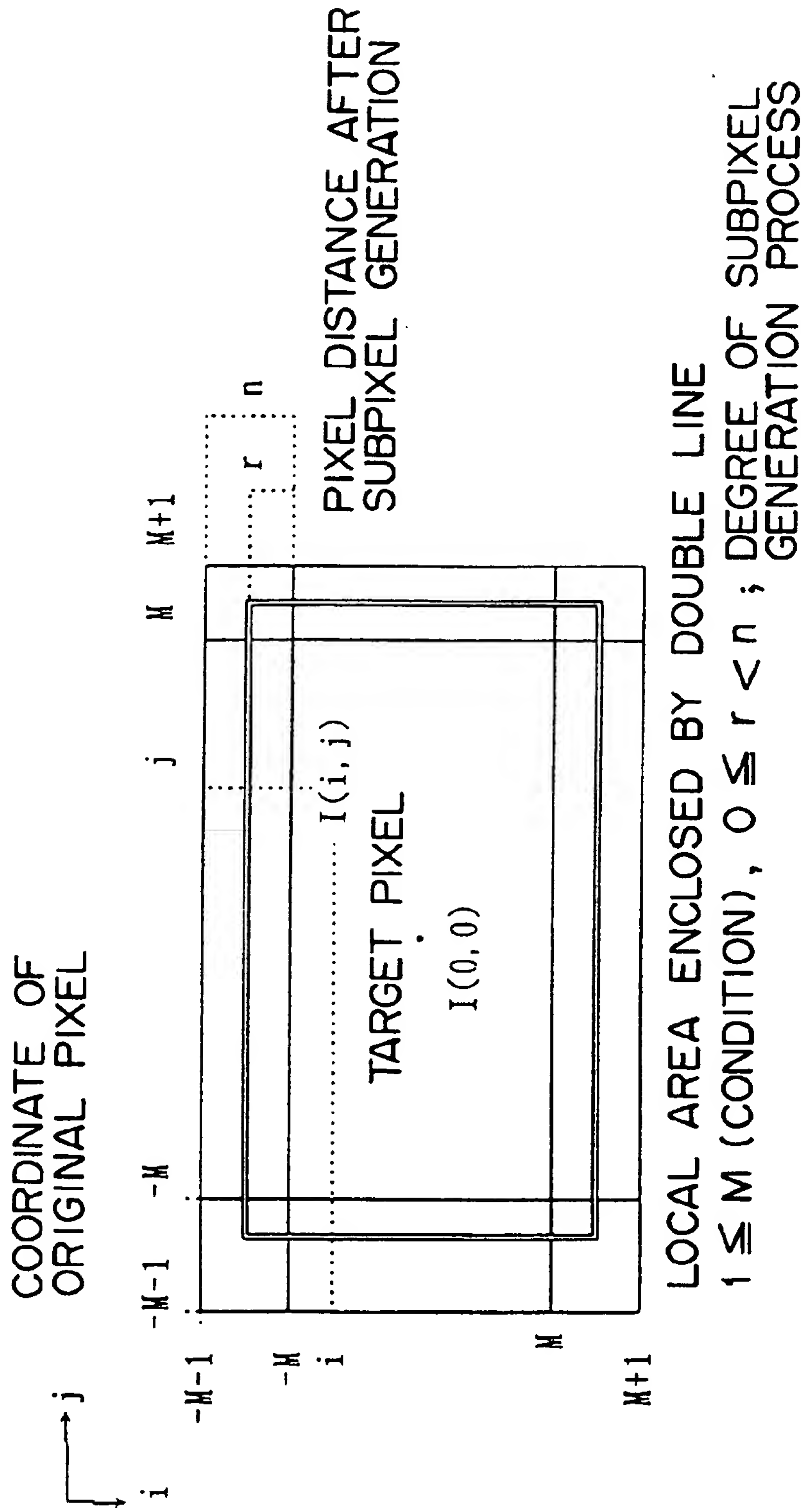
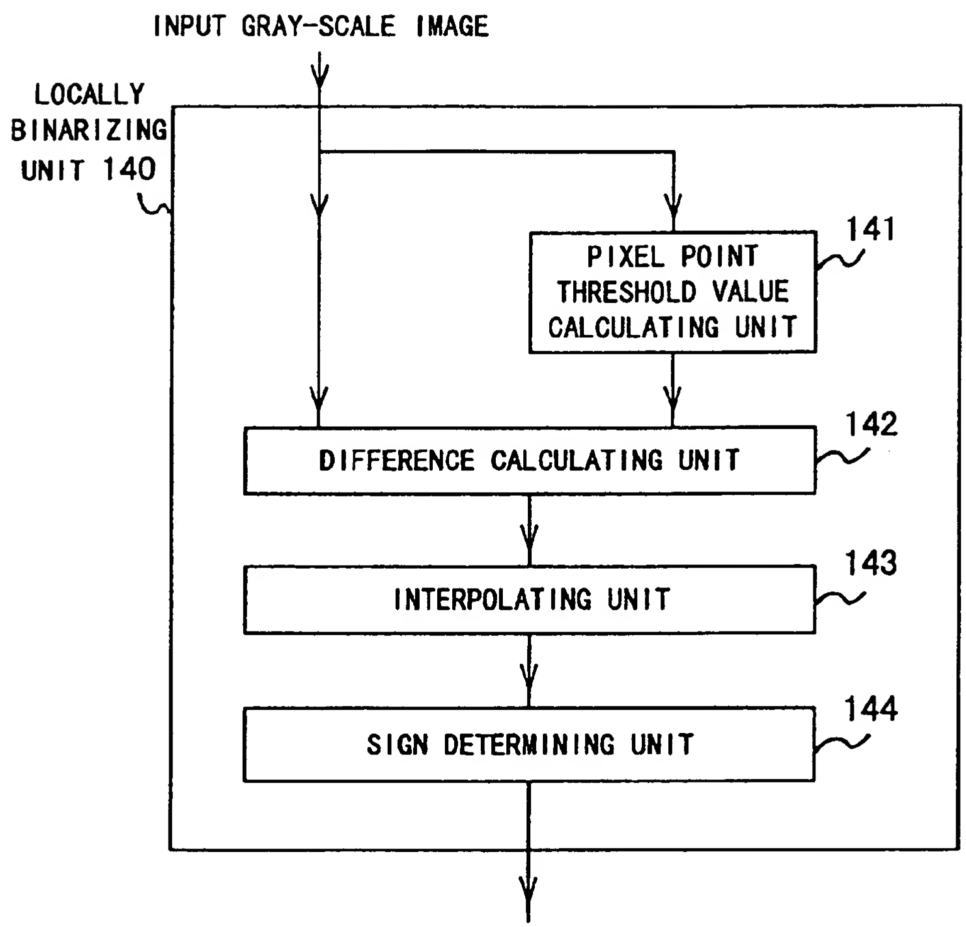

INPUT GRAY-SCALE IMAGE
80
SUBPIXEL PROCESSING UNIT
LOCALLY BINARIZING UNIT 120
121
LOCAL AREA ORIGINAL PIXEL BOUNDARY DETERMINING UNIT
126
LOCAL AREA SPECIFYING UNIT
122
ORIGINAL PIXEL BOUNDARY PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
123
ORIGINAL PIXEL NON-BOUNDARY PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
124
INTERPOLATING UNIT
125
COMPARING UNIT

FIG. 13



COORDINATE OF ORIGINAL PIXEL
PIXEL DISTANCE AFTER SUBPIXEL GENERATION
-M-1
-M
j
M
M+1
r
n
i
I(i, j)
TARGET PIXEL
I(0,0)
LOCAL AREA ENCLOSED BY DOUBLE LINE
1 ≦ M (CONDITION), 0 ≦ r < n ; DEGREE OF SUBPIXEL GENERATION PROCESS

FIG. 14


INPUT GRAY-SCALE IMAGE
LOCALLY BINARIZING UNIT 140
PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
141
DIFFERENCE CALCULATING UNIT
142
INTERPOLATING UNIT
143
SIGN DETERMINING UNIT
144

F I G. 1 5

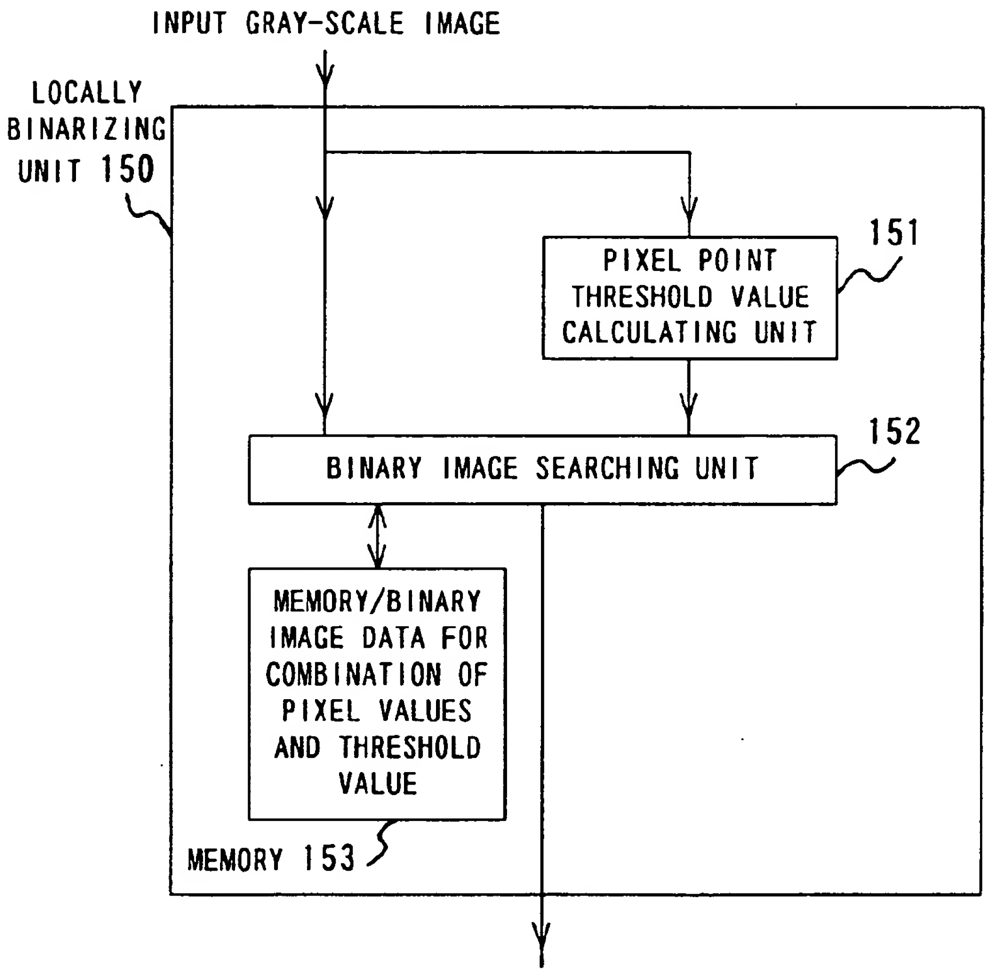

INPUT GRAY-SCALE IMAGE
LOCALLY BINARIZING UNIT 150
PIXEL POINT THRESHOLD VALUE CALCULATING UNIT
151
BINARY IMAGE SEARCHING UNIT
152
MEMORY/BINARY IMAGE DATA FOR COMBINATION OF PIXEL VALUES AND THRESHOLD VALUE
MEMORY 153

FIG. 16

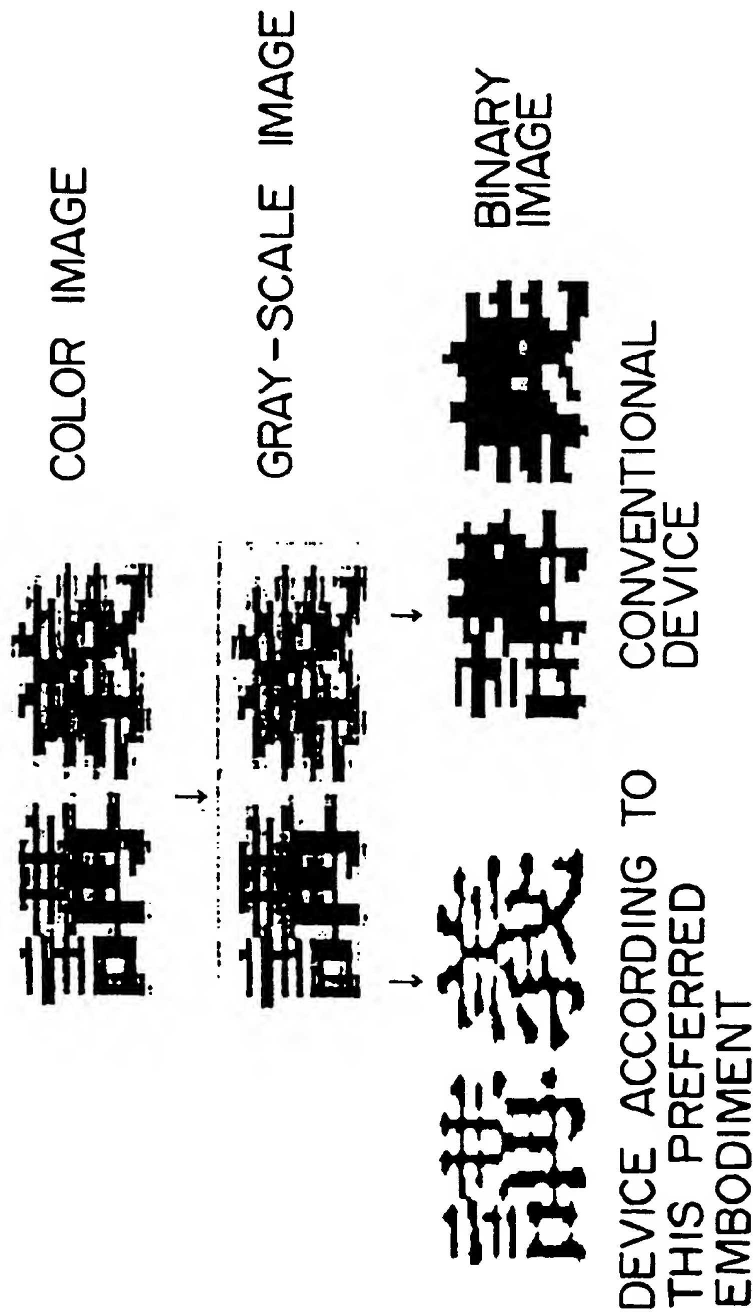

COLOR IMAGE
GRAY-SCALE IMAGE
BINARY IMAGE
CONVENTIONAL DEVICE
DEVICE ACCORDING TO THIS PREFERRED EMBODIMENT

FIG. 17

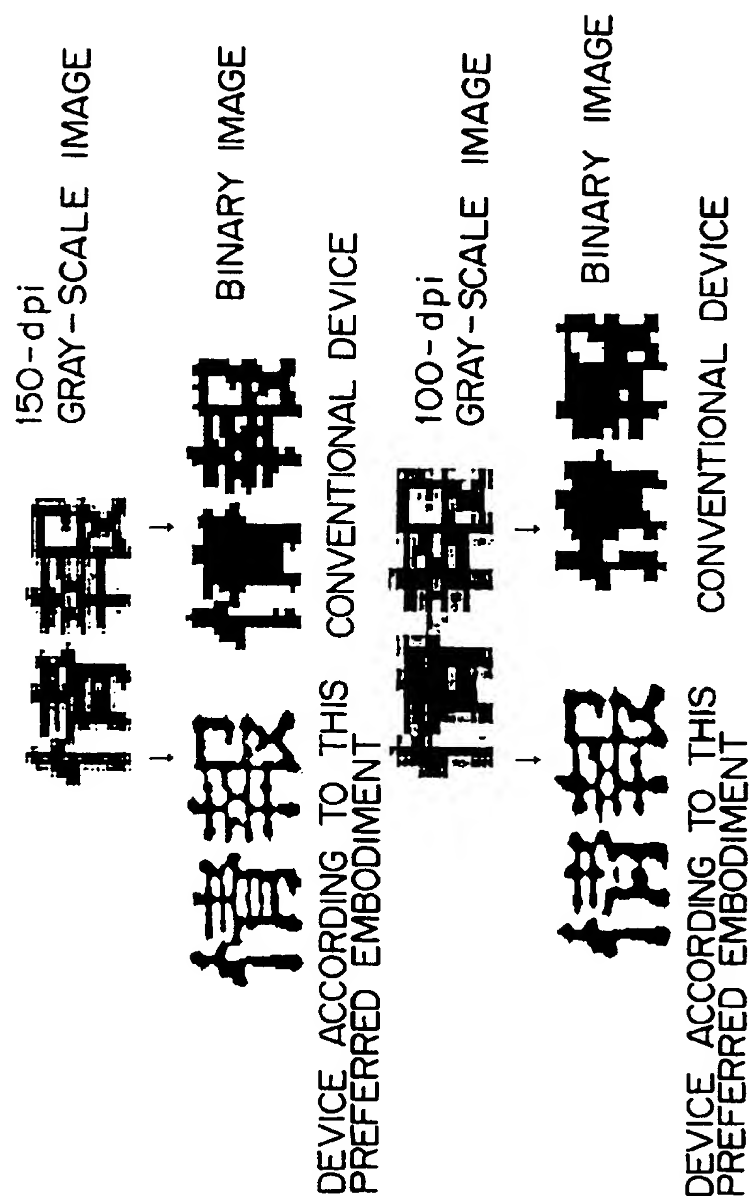

150-dpi
GRAY-SCALE IMAGE
BINARY IMAGE
DEVICE ACCORDING TO THIS PREFERRED EMBODIMENT
CONVENTIONAL DEVICE
100-dpi
GRAY-SCALE IMAGE
BINARY IMAGE
DEVICE ACCORDING TO THIS PREFERRED EMBODIMENT
CONVENTIONAL DEVICE

FIG. 18

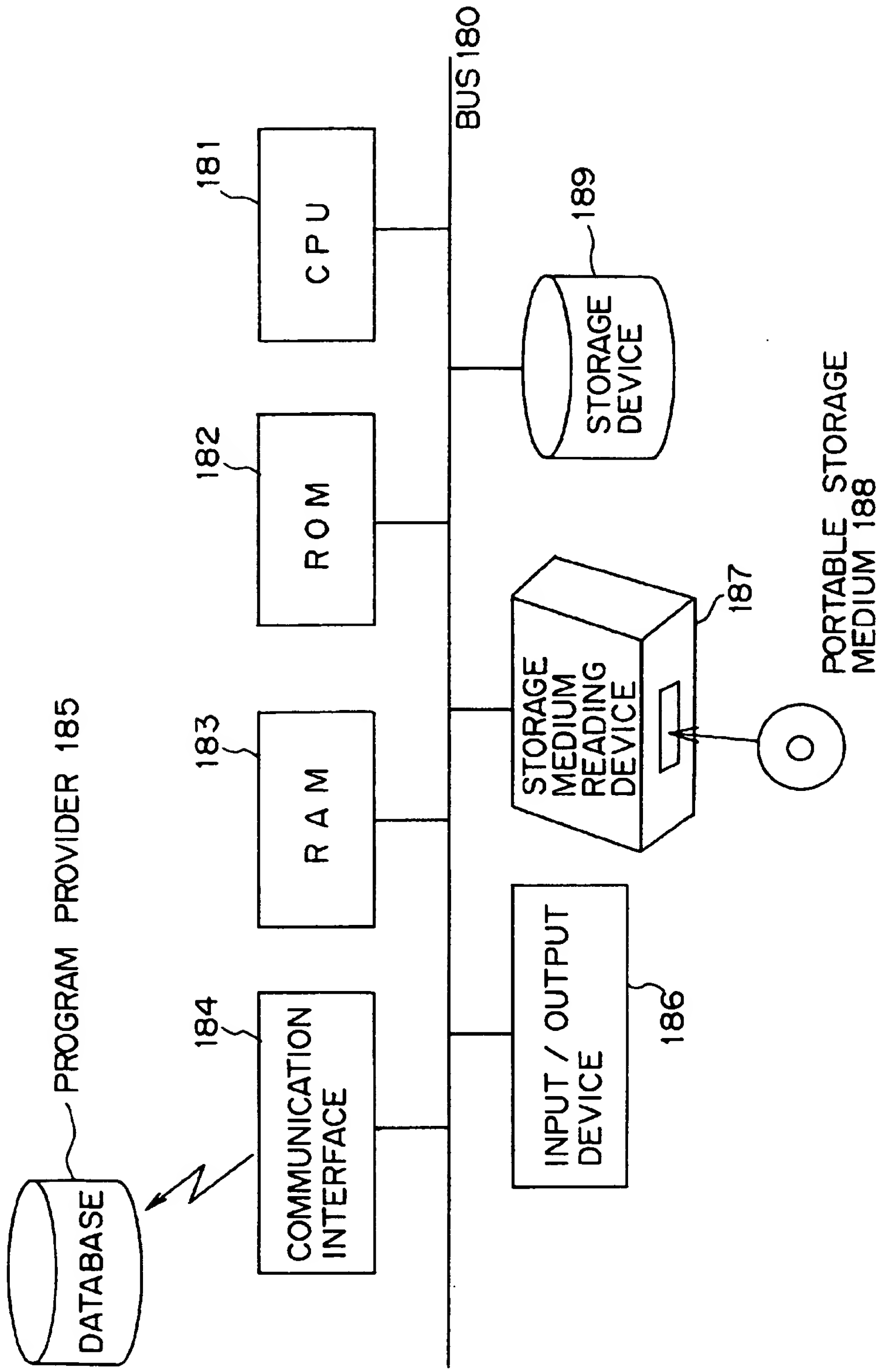

BUS 180
181
CPU
182
ROM
183
RAM
184
COMMUNICATION INTERFACE
PROGRAM PROVIDER 185
DATABASE
189
STORAGE DEVICE
187
STORAGE MEDIUM READING DEVICE
PORTABLE STORAGE MEDIUM 188
INPUT / OUTPUT DEVICE
186

FIG. 19

DEVICE, METHOD AND STORAGE MEDIUM FOR RECOGNIZING A DOCUMENT IMAGE

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to an image recognizing device.

2. Description of the Related Art

With the popularization of personal computers and the arrangements of networks, the number of electronic documents has been growing in recent years. However, the main medium of information distribution is still a paper document, and an enormous number of paper documents currently exist. Accordingly, a document image recognizing device intended to convert a paper document into an electronic document, and to edit a conversion result has been increasingly demanded. The document image recognizing device is a device which uses a document image as an input, and performs a coding process by recognizing characters, etc. included in the document image. Currently, there are products which use a binary document image as an input. The binary document image is a document image where image data of a character is represented, for example, by "1", while the image data of the background is represented by "0".

Recently, an image filing device has become popular, and also the demand for inputting an image filed by the image filing device to a document image recognizing device, and for recognizing the image has been increasing. Especially, the number of gray-scale or color documents including photographs has been growing. Therefore, the demand for recognizing not only binary documents but also gray-scale or color documents has been on the rise.

A color document image recognizing device recognizes also a color or gray-scale document image. A conventional color document image recognizing device obtains a binary image by binarizing each brightness component with a predetermined threshold value, and recognizes the obtained binary image, if an input document image is not a binary image but a gray-scale or color image.

FIGS. 1A and 1B respectively show the configuration of the conventional color document image recognizing device and an extended color text image.

In FIG. 1A, a document image inputting unit 170 is a unit for inputting a document image, and is typically implemented as a scanner, etc. For a color document, parameters such as a color parameter, a brightness parameter, etc. are assigned to respective pixels by illuminating the document, receiving a reflected light, and analyzing the received light. If the document to be scanned is a gray-scale document, the light reflected from the document is analyzed, the information about the level of brightness is obtained, and this information is assigned to each pixel. At this time, all of the pixels of the gray-scale document are detected to be black-and-white, which is set as the color parameters of each of the pixels.

If an input image is a color image, a brightness image extracting unit 171 extracts a brightness component for each pixel, and outputs a brightness image which is a gray-scale image to a predetermined threshold value binarizing unit. If the input image is a gray-scale image, the color parameters of all of the pixels are set to be black-and-white. Therefore, the gray-scale image resultant from the process of the brightness image extracting unit 171 will become the image data having the same brightness data as that of the input gray-scale image, in principle. This is because only the parameters related to the hue are removed from all the color parameters of the processed gray-scale image.

If a gray-scale image which is a brightness image is input, a predetermined threshold value binarizing unit 172 obtains a binary image by binarizing the gray-scale image with a predetermined threshold value. This threshold value is a value which is externally determined and input. Hereinafter, a gray-scale image fundamentally indicates not the gray-scale image with color parameters set to black-and-white, which is scanned by the document image inputting unit 170, but the brightness image resultant from the process performed by the brightness image extracting unit 171. That is, the gray-scale image is defined to have not color but brightness parameters. Even if the gray-scale image has color parameters, only brightness parameters are substantially valid for recognizing the image if the color parameters are set to black-and-white for all of the pixels. Accordingly, image recognition can be made also by using such a gray-scale image.

A binary image recognizing unit 173 recognizes a binary image. That is, this unit recognizes characters by obtaining the features of the document image which is binarized by the predetermined threshold value binarizing unit 172, and replaces the characters with the codes which are internally used by a computer and correspond to the recognized characters.

A recognition result outputting unit 174 outputs the result of the character recognition made by the binary image recognizing unit 173, that is, the file which is restructured into a code sequence representing the characters of a document image.

The conventional color document image recognizing device has a disadvantage that recognition accuracy is low.

Since the amount of data of each pixel of a color document image is 8 times that of a binary image, the color document image is input to a color document image recognizing device at a resolution lower than that of the binary image in order to reduce an amount of processing time of an image input device such as a scanner, etc., and a capacity of a memory used for filing an image, etc.

FIG. 1B illustrates an expanded low-resolution color text image.

This figure shows a monochromatic image of a 150-dpi text image in full color. With a color display, many colors can be identified on the periphery of the characters, and it seems difficult to identify an area by extracting the same color.

The conventional document image recognizing device is fundamentally designed to have an input of a binary image with a small amount of data, and assumes the document image of a standard resolution of approximately 400 dpi. Accordingly, if a color document image of 150 or less dpi is input, the conventional device converts the document image into a binary image with the resolution equal to or less than 150 dpi, and recognizes the converted image. Therefore, the device cannot recognize the image with sufficient accuracy.

SUMMARY OF THE INVENTION

An object of the present invention is to provide a high-speed document image recognizing device which implements high recognition accuracy.

An image recognizing device according to the present invention comprises: an image converting unit for convert-

ing an input document image into a gray-scale image if the input document image is a color image, and for newly outputting a gray-scale image if the input document image is a gray-scale image; a variable resolution binarizing unit for converting the gray-scale image into a binary image with a higher resolution according to the resolution of the gray-scale image; and a unit for recognizing the binarized image.

An image recognizing method according to the present invention comprises the steps of: (a) converting an input document image into a gray-scale image if the input document image is a color image, and for newly outputting a gray-scale image if the input document image is a gray-scale image; (b) converting the gray-scale image into a binary image with a higher resolution according to the resolution of the gray-scale image; and (c) recognizing the binarized image.

motivation

According to the present invention, a gray-scale image obtained by converting a color document image or an input gray-scale image is converted into the image data having a suitably higher resolution according to the resolution of the gray-scale image, thereby binarizing the image without losing the information about brightness levels of a gray scale. Therefore, characters appearing in a binary image can be prevented from being defaced, thereby implementing character recognition with higher accuracy.

Furthermore, a drawing area including characters, etc. is roughly extracted from a document image, and a binarization process according to the present invention is performed only for the extracted area, thereby improving the processing speed.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A shows the configuration of a conventional color document image recognizing device;

FIG. 1B shows an expanded color text image;

FIG. 2 is a block diagram showing the principle of a color document image recognizing device according to a preferred embodiment;

FIGS. 3A and 3B exemplify the configurations of a variable resolution binarizing unit;

FIG. 4 is a schematic diagram explaining the principle of the process for generating subpixels;

FIG. 5 exemplifies the table for specifying the number and the resolution of subpixels to be generated according to the resolution of an input gray-scale image;

FIG. 6 exemplifies an additional configuration of the variable resolution binarizing unit;

FIG. 7 is a schematic diagram explaining the method for setting a global threshold value used for a rough extraction process;

FIGS. 8A and 8B exemplify the further configurations of the variable resolution binarizing unit;

FIG. 9 exemplifies the details of a first configuration of a locally binarizing unit;

FIG. 10 exemplifies the details of a second configuration of the locally binarizing unit;

FIG. 11 exemplifies a third configuration of the locally binarizing unit;

FIG. 12 exemplifies a fourth configuration of the locally binarizing unit;

FIG. 13 exemplifies a fifth configuration of the locally binarizing unit;

FIG. 14 is a schematic diagram explaining the method for calculating a local average value and a local square average value from original pixel values;

FIG. 15 exemplifies a sixth configuration of the locally binarizing unit;

FIG. 16 exemplifies a seventh configuration of the locally binarizing unit;

FIG. 17 is a schematic diagram exemplifying the processing up to the process for binarizing a color or gray-scale image with the process according to the preferred embodiment (No. 1);

FIG. 18 is a schematic diagram exemplifying the processing up to the process for binarizing a color or gray-scale image with the process according to the preferred embodiment (No. 2); and

FIG. 19 is a block diagram explaining the configuration of the hardware required for implementing the preferred embodiment as software.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

According to the present invention, an input image is converted into a binary image with a higher resolution according to the resolution of the input image in order to overcome the above described problems of the conventional device.

Even if the resolution of an input image is low, a human being can recognize the image if it is a color or gray-scale image. However, even a human being has difficulty in recognizing an image which is binarized by the conventional device. This is because the information about the brightness components of the original image is lost. To facilitate the recognition of the binarized image, the information about the brightness components of the image must be reflected in the binarized image. For the implementation of the reflection, it is effective to increase the resolution of the binarized image.

If an input document image is a color image, it is converted into a gray-scale image and further converted into a binary image with the resolution according to that of the input image. The converted binary image is recognized and the characters are converted into electronic codes.

As a specific means for converting a gray-scale image into a binary image having the resolution according to that of an input image, a method for obtaining a binary image after performing the subpixel generation process which increases the number of pixels included in a gray-scale image by interpolating the values of the pixels included in the gray-scale image, is considered. As a specific method of the subpixel generation process, a linear interpolation method between pixel values can be cited.

Additionally, the method for extracting a character portion and its periphery from an entire image, for suitably generating a threshold value within the extracted partial image, and for performing character recognition, can be cited as a method for improving the recognition ratio of a color or gray-scale image. With this method, the result of the binarization process allows a character to be clearly shaped by reducing the light and shade of an area having a higher level of brightness in the background, that is, noise components rather than by preventing the information of brightness components from being lost.

FIG. 2 is a block diagram showing the principle of a color document image recognizing device according to a preferred embodiment.

A document image inputting unit 10 is a unit corresponding to a scanner in a similar manner as in the above described conventional technique, and is intended to illuminate a color

or gray-scale document (including parameters related to hue), and to capture the document as an image.

A brightness image extracting unit 11 is a unit for extracting only brightness components from the image input by the document image inputting unit 10, and for generating a gray-scale image (which does not include parameters related to hue). The gray-scale image which is the output of the brightness image extracting unit 11 is input to a variable resolution binarizing unit 12 to be described next.

The variable resolution binarizing unit 12 (roughly) extracts a partial area including characters from the input gray-scale image having a low resolution, and provides a binary image where characters are easily recognized to a binary image recognizing unit 13 at a succeeding stage by generating subpixels between original pixels within the gray-scale image and interpolating the information about brightness components.

The binary image recognizing unit 13 performs character recognition based on the binary image passed from the variable resolution binarizing unit 12, and performs the process for replacing characters of an image with electronic codes.

A recognition result outputting unit 14 receives the document file as an electronic code sequence from the binary image recognizing unit 13, stores the document file onto a storage medium such as a hard disk, etc., and outputs the document file as a recognition result to a display monitor.

Since the configurations of the document image inputting unit 10, the brightness image extracting unit 11, the binary image recognizing unit 13, and the recognition result outputting unit 14 are similar to those of the conventional technique, their detailed explanations are omitted here. That is, even if they are omitted, a skilled artisan can easily understand the configurations of the document image inputting unit 10, the binary image recognizing unit 13, and the recognition result outputting unit 14, and can actually use them in a current situation where devices and software for recognizing a black-and-white document are commercialized. Additionally, the skilled artisan can easily understand also the configuration of the brightness image extracting unit 11, because this unit is intended to convert a color image, etc. into a gray-scale image, and, at present, a color image is converted into a black-and-white image and telecast. Therefore, its explanation is omitted here.

Accordingly, the present invention is characterized in that a predetermined threshold value binarizing unit 172 is replaced with a variable resolution binarizing unit 12. The explanation to be provided below will refer to the details of the variable resolution binarizing unit 12. The variable resolution binarizing unit 12 converts a gray-scale image into the binary image having the resolution according to that of an input image, or binarizes a gray-scale image after roughly extracting a portion including characters from a gray-scale image, so that the image can be used for character recognition. Additionally, the resolution conversion and the rough extraction may be performed at the same time.

FIGS. 3A and 3B are block diagrams exemplifying the configurations of the variable resolution binarizing unit.

As a first example of the configuration of the variable resolution binarizing unit, subpixels are generated between original pixels of a gray-scale image by a subpixel processing unit 20, the resolution is increased according to the resolution of the gray-scale image which is an input image, and then the conventional process for binarizing an image with a predetermined threshold value is performed by a predetermined threshold value binarizing unit 21, as shown in FIG. 3A.

applicable to R, G, or B pixels since each pixel values in eq. to "gray" levels. ("gray" = R, G, or B)

Here, a predetermined threshold value is defined to be used for a single document image. For example, only one threshold value is used for binarizing a 1-page document image.

The process for generating subpixels is a process for subdividing the space between original pixels of a gray-scale image whose recognition ratio is not likely to be improved if it is binarized unchanged, and for generating virtual pixel data in the original data. Although its details will be described later, in short, the pixel value (the level of brightness) of a subpixel is obtained by interpolating the levels of brightness of original pixels of an input gray-scale image. Typical of the interpolation method is a linear interpolation method.

As a second example of the configuration of the variable resolution binarizing unit, a locally binarizing unit 23 shown in FIG. 3B may be arranged instead of the predetermined threshold value binarizing unit 21 shown in FIG. 3A, as illustrated in FIG. 3B. The locally binarizing unit 23 sets the local area where the process is to be performed for each pixel included in a drawing area within an entire gray-scale document image, and obtains a binary image by binarizing the image with the threshold value generated by using the pixel data within the local area.

That is, the threshold value of the level of brightness is obtained in a local range (local area such as a regular square area centering around a target pixel), and the binarization process is performed. That is, the binarization process is performed so that the pixel with the level of brightness which is equal to or lower than the threshold value is black (the level of brightness is set to, for example, "1"), and the pixel with the level of brightness which is equal to or higher than the threshold value is white (the level of brightness is set to, for example, "0"). The threshold value of a local area is defined, for example, with the method using the linear combination of an average pixel value, a standard deviation value, and a variance. The threshold value is defined, for example, as follows. In the following equation, a local binarization parameter is a constant, and an optimum threshold value is obtained by suitably setting this parameter. Note that the gray scale is used almost similar to the brightness in the following equation.

(threshold value)=(average gray scale)+(local binarization parameter)×(gray-scale standard deviation value)

FIG. 4 is a schematic diagram explaining the principle of the process for generating subpixels.

In this figure, black circles indicate original pixels of a gray-scale image, while white circles indicate subpixels. Additionally, IA through ID indicate the levels of the gray scale of the original pixels A through D. Furthermore, I1 through I5 indicate the levels of gray scale of the subpixels 1 through 5 to be obtained from the levels of gray scale of the original pixels with the interpolation process.

As shown in this figure, if subpixels are generated with the linear interpolation within an area enclosed by the four original pixels, it is first determined how many subpixels will be arranged between the original pixels. Next, subpixels are positioned at regular intervals according to the number of subpixels to be arranged. Then, the levels of gray scale are assigned to the respective subpixels by interpolating the levels of gray scale of the original pixels.

Considered below is the case where the subpixels are arranged in the area enclosed by the original pixels A through D based on the assumption that "p" and "q" are numbers which are larger than "0" and smaller than "1". A

subpixel 1 is arranged at the position represented by "p:1−p" in a straight line AB linking the original pixels A and B. In this case, the levels of gray scale of the subpixel 1 is obtained from the levels of gray scales IA and IB of the original pixels A and B with the linear interpolation according to the following equation.

$$I1=p*IB+(1-p)*IA$$

Similarly, the levels of gray scales of the subpixels 2 through 4 are obtained as follows.

$$I2=q*IC+(1-q)*IA$$

$$I3=q*ID+(1-q)*IB$$

$$I4=p*ID+(1-p)*IC$$

Additionally, the level of gray scale I5 of the subpixel 5 included in the area enclosed by the original pixels A through D can be obtained according to the following equation.

$$I5=p*q*ID+p*(1-q)*IB+q*(1-p)*IC+(1-p)*(1-q)*IA$$

The above described calculation is made for all of the subpixels arranged between the original pixels, so that the process for generating subpixels is completed. The obtained levels of gray scale are stored with a method similar to that of original pixel data, along with the corresponding positions of the subpixels.

FIG. 5 exemplifies a table for specifying the number of subpixels to be generated, according to the resolution of a gray-scale image which is an input image.

The process for generating subpixels between original pixels must be performed at the beginning of the calculation process explained by referring to FIG. 4. Although the number of subpixels to be generated is arbitrary, it must be set so that sufficient character recognition accuracy can be obtained from the image for which subpixels are generated and which is binarized. Accordingly, there is no equation for uniquely determining the number of subpixels to be generated according to the resolution of an input image. The number of subpixels to be generated must be determined based on experimental data to some extent.

In FIG. 5, the resolution of an input image is stored in the left column in correspondence with a subpixel generation parameter in the right column. For example, if the resolution of an input image is 100 dpi, subpixels are generated to form a grid of 6×6 for each original pixel. If the resolution of an input image is 150 dpi, subpixels are generated in order to form a grid of 3×3 for each original pixel.

In this way, the above described subpixel processing unit stores the table shown in FIG. 5. When a gray-scale image is input, the subpixel processing unit obtains the number of subpixels to be generated according to the resolution of the input image, and determines the positions of the subpixels. Next, the subpixel processing unit obtains the levels of gray scale of the respective subpixels from the levels of gray scale of the original pixels with the interpolation process by using the calculation method explained by referring to FIG. 4, and generates the image data having a high resolution according to the resolution of the input image.

FIG. 6 is a block diagram exemplifying a further configuration of the variable resolution binarizing unit.

The configuration shown in FIG. 3B relatively requires a considerable amount of time in order to locally binarize an entire image area. In the meantime, with the configuration shown in FIG. 6, a drawing area roughly extracting unit may perform the global process for recognizing as the periphery of characters the image portion whose pixel values are lower than a global threshold value set for the pixel values of a gray-scale image, and for roughly extracting the drawing area. Then, a local binarization process may be performed.

That is, the drawing area roughly extracting unit 50 extracts the portion including characters and its periphery from a document image, and the locally binarizing unit binarizes the gray-scale image within the extracted area. The rough extraction is performed by totaling the number of pixels having one level of gray scale or brightness of all of the pixels of a document image, as will be described later. If the number of pixels is large in a portion where the level of brightness is high, it indicates the background of a document image. If the number of pixels is large in a portion where the level of brightness is low, it indicates the portion including a character of the document image. A threshold value is set to a value in the portion where the number of pixels is small, which corresponds to the middle of the two peaks. This threshold value setting is similar to that of the binarization process. Note that, however, the position at which the threshold value is set is slightly shifted to a higher level of brightness. If the pixels included in a document image are extracted with such a threshold value (global threshold value), the extracted portion will include a character and its periphery. The locally binarizing unit 51 binarizes the image with another threshold value for the roughly extracted area, so that noise components caused by the light and shade of the background of the document image can be removed. Consequently, the binary image which is easier to be recognized can be obtained.

Specifically, the global threshold value is determined with the linear sum of an average pixel value, a standard deviation value, and a variance. Or, it can be determined as follows. A global process parameter which will appear in the following equation is a constant.

(global threshold value)=(an average of all pixel values)+(global process parameter)×(standard deviation value of all pixel values)

The position of the global threshold value is adjusted to the position where the rough extraction can be most effectively made by making an adjustment with the global process parameter.

FIG. 7 is a schematic diagram explaining how to set a global threshold value used for the rough extraction process.

When the rough extraction process is performed, as shown in this figure, the levels of brightness are obtained from all of the pixels structuring an entire document image, and the statistics of the frequency at which a pixel having a particular level of brightness appears in the document is collected in a similar manner as in the case where the predetermined threshold value is determined for the entire document image. In this figure, the appearance frequency of the pixel against the level of brightness forms a gentle curve. However, this is illustrated for ease of explanation about the threshold value setting method. If the statistical process representing the frequency at which a pixel having a certain level of brightness appears is actually performed with a device, the result will be a histogram.

In FIG. 7, frequency peaks are formed in the respective portions where the levels of brightness are low and high. The portion where the level of brightness is high is the background of a document image, and this level of brightness is the level of brightness of the paper on which the document is created. In the meantime, the portion where the level of brightness is low is an area where a character or a graphic,

etc. is drawn. If the frequencies at which pixels appear are classified depending on the levels of brightness, the pixels are grouped into two major groups such as the pixels included in the drawing area and the pixels included in a background. Therefore, the rough extraction process can be performed by setting the threshold value of the level of brightness to almost the middle of the two peaks, and extracting the pixels having the levels of brightness which are lower than this threshold value from the document image.

If a document is structured by the reverse video of a black-and-white image, the area where the two peaks are drawn and the background become reversed. Accordingly, the pixels having levels of brightness which are higher than a preset threshold value, are output as a drawing area.

If an entire document is binarized with a predetermined threshold value, the threshold value is set to a value corresponding to almost the middle of the two peaks, that is, the value at the bottom of the frequency curves. This is because the character recognition cannot be made with high accuracy if the pixels of the background are also captured. When the rough extraction process is performed, however, an extracted area which includes the periphery of a character is more convenient to set a threshold value in the binarization process. Therefore, the threshold value is set to the value which is slightly shifted from the bottom toward the peak of the background. In this way, the area where a character, etc. is drawn and its periphery are extracted.

FIGS. 8A and 8B are block diagrams exemplifying further configurations of the variable resolution binarizing unit.

If the configurations shown in FIGS. 3A and 6 are combined, the global process for roughly extracting a drawing area is performed after the subpixel generation process is performed for a gray-scale image. Then, the local binarization process is performed for each of the pixels included in the drawing area. As a result, the processes can be performed at a higher speed than that implemented by the configuration shown in FIG. 3A, and at the same time, the recognition accuracy is improved more than that available from the configuration shown in FIG. 6.

With the configuration shown in FIG. 8A, the subpixel generation process is initially performed by a subpixel processing unit 70, and then a document image is roughly extracted by a drawing area roughly extracting unit 71. With the subpixel generation process, the information about the level of brightness of a gray-scale image can be prevented from being lost, or rather, the information about the level of brightness is included. As a result, the accuracy of the binarization process can be improved. Furthermore, performing the rough extraction process eliminates the need to target the whole of a document image, thereby reducing the number of pixels to be processed and improving the processing speed of the locally binarizing unit.

The subpixel processing unit 70 performs the subpixel generation process for the whole of a document image. The drawing area roughly extracting unit 71 performs the rough extraction process by also targeting subpixels. Accordingly, the processing speed and the accuracy can be improved further than that implemented in the case where only the rough extraction process is performed. However, the subpixel generation process must be performed for the whole of the document image in this case. Therefore, the amounts of data handled by the subpixel processing unit 70 and the drawing area roughly extracting unit 71 increase, which leads to the slowdown of the processing speed and the insufficiency of the capacity of a memory storing data, although the processing speed of the locally binarizing unit 72 can be improved.

FIG. 8B is a block diagram exemplifying the configuration which improves the processing speed implemented by the configuration shown in FIG. 8A.

The configuration shown in FIG. 8B is obtained by reversing the processing order of the subpixel processing unit 70 and the drawing area roughly extracting unit 71, which are included in the configuration shown in FIG. 8A. Because the process of the drawing area roughly extracting unit 73 is performed for the original gray-scale image whose number of pixels is not increased with the subpixel generation process, the amount of data handled by the drawing area roughly extracting unit 73 is reduced and the processing speed is made faster.

Additionally, it is sufficient for the subpixel processing unit 74 to also perform the subpixel generation process, not for the whole of an input gray-scale image, but only for the area extracted by the drawing area roughly extracting unit 73. Accordingly, the number of pixels to be handled is reduced much more than that in the case where the subpixel generation process is performed for the whole of the input gray-scale image, in the configuration shown in FIG. 8A. Consequently, the capacity of the memory storing data can be decreased, and at the same time, the processing speed can be improved.

Furthermore, since the amount of throughput of the locally binarizing unit 75 is approximately the same as that of the corresponding unit shown in FIG. 8A, the processing speed on the whole can be improved, and also the amount of required hardware resources such as a memory, etc. can be reduced.

Generally, with the configurations (shown in FIGS. 3A, 8A, and 8B) for performing the local binarization process after the subpixel generation process, the threshold value of the local binarization process is calculated from the subpixel value at the interpolation point (subpixel). However, the subpixel value at the interpolation point is obtained from the original pixel values, the local binarization process can be performed according to the equation obtained by assigning the equation for obtaining a subpixel value at an interpolation point from original pixel values, to the equation for obtaining a local threshold value from the subpixel value at the interpolation point.

FIG. 9 is a block diagram exemplifying the details of the configuration of the locally binarizing unit.

Because the contents of the subpixel generation process performed by a subpixel processing unit 80 are the same as those explained by referring to FIG. 4, the explanation is omitted here. A locally binarizing unit 81 is composed of a threshold value calculating unit 82 and a comparing unit 83. The threshold value calculating unit 82 directly obtains the threshold value of the local binarization process from a gray-scale image. The comparing unit 83 makes a comparison between the threshold value and the value at the point where a subpixel is generated, which is obtained by the subpixel processing unit 80, and outputs the result of the binarization process.

That is, the information about a pixel point of an input gray-scale image is directly input to the threshold value calculating unit 82. Furthermore, the threshold value calculating unit 82 sets a local area by targeting a particular pixel point, statistically classifies the level of gray scale (the level of brightness) at each pixel point within the local area as explained by referring to FIG. 7, and sets a threshold value which allows the distinction between the pixel having the level of brightness which represents a background, and the pixel having the level of brightness which represents a drawing area, to be made as definite as possible. The rough

extraction process is explained by referring to FIG. 7, and the threshold value which is shifted to the level of brightness, which represents a background, is described to be used. If the binarization process is performed, the threshold value is set to the value corresponding to the bottom of the frequency curves shown in the middle of FIG. 7. With such a threshold value setting, the distinction between the pixel structuring a character and the pixel structuring a background can be definitely made. As a result, a binary image which is easier to be recognized can be obtained.

The threshold value calculated by the threshold value calculating unit 82 is input to the comparing unit 83. The subpixel processing unit 80 generates the information about each pixel point of the gray-scale image for which the subpixel generation process is performed, and about the level of brightness at each pixel point, and inputs to the comparing unit 83 the information about the level of brightness of the pixel point targeted by the threshold value calculating unit 82. The comparing unit 83 then makes a comparison between the level of brightness at each pixel point and the threshold value. Here, assume that the level of brightness at a pixel point, which is higher than the threshold value, is set to "0", while the level of brightness at the pixel point, which is lower than the threshold value, is set to "1". The pixel point to be targeted is sequentially changed, and the above described process is repeatedly performed, so that a binary image corresponding to the input gray-scale image can be obtained. The pixel point referred to in the above explanation includes both an original pixel of an input gray-scale image and a subpixel generated by performing the subpixel generation process. If the pixel point is used to indicate an original pixel of an input gray-scale image, it is hereinafter referred to as an original pixel point. With the configuration shown in FIG. 9, the binarization process is performed for the whole of image data whose number of pixels is increased by performing the subpixel generation process for the whole of the input gray-scale image.

FIG. 10 shows the details of the second configuration of the locally binarizing unit.

In this figure, the local threshold value after the subpixel generation process is obtained only at each original pixel point of a gray-scale image. The local threshold value at an interpolation point (a subpixel point) is obtained by interpolating the local threshold values at the original pixel points of the gray-scale image. Namely, with the configuration shown in FIG. 9, both subpixel points and original pixel points of a gray-scale image are handled in a similar manner, and the threshold value for the local binarization process is calculated. However, with the configuration shown in FIG. 10, only the threshold value at an original pixel point is calculated, and the threshold value at a subpixel point is obtained with the interpolation process. Actually, the interpolation process explained by referring to FIG. 4 is not performed for the level of brightness at each original pixel point. The threshold value at a subpixel point is obtained by interpolating threshold values obtained at original pixel points, similar to the level of brightness.

That is, a pixel point threshold value calculating unit 91 obtains the local threshold value after the subpixel generation process only at an original point of a gray-scale image. An interpolating unit 92 obtains the local threshold value at an interpolation point (subpixel) by interpolating the local threshold values at the pixel points of the gray-scale image. The comparing unit 83 makes a comparison between the pixel value at a pixel point, which is obtained by the subpixel processing unit 80, and the threshold value obtained by the interpolating unit 92, and outputs the result of the binarization process. Such a binarizing process is performed for all of the pixel points obtained by the subpixel processing unit 80, thereby obtaining a binary image.

The method for obtaining the local threshold value at an original pixel point of a gray-scale image, which is executed by the pixel point threshold value calculating unit 91, includes the following methods.

(1) Obtaining the local threshold value at an original pixel point of a gray-scale image by using the value of the subpixel obtained by interpolating original pixel points of the gray-scale image.

(2) Obtaining the local threshold value by substituting the equation for obtaining the subpixel value at the interpolation points from the original pixel values into the equation for obtaining the local threshold value from the subpixel value at the interpolation points.

(3) Obtaining the subpixel value from the linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, from an equation obtained by substituting an equation for obtaining a subpixel value from original pixel values, to an equation for obtaining the average value and the square average value from subpixel values.

The configuration implementing the method (1) is shown in FIG. 10.

The configuration implementing the method (2) is shown in FIG. 11. The pixel point threshold value calculating unit uses only the pixel values of a gray-scale image.

The method (3) is applicable if the local threshold value is the linear combination of an average pixel value, a standard deviation value, and a variance within a local area. The configuration implementing the method (3) is shown in FIG. 12. The variance is obtained from the average pixel value and the square average value. The standard deviation value is obtained from the variance. The specific example of the equation will be provided next.

variance=an average of square values−the square of an average value standard deviation=√(variance)

If the local area recognizes original pixels as its boundary, the equations for obtaining the local threshold value and the average values are simplified, which leads to a speeding-up of the processing. Therefore, the method using these expressions may be available.

Additionally, the locally binarizing unit may comprise a local area specifying unit, by which the process may be performed by making a distinction between the case where a local area does not recognize original pixels as its boundary, and the case where the local area recognizes the original pixels as its boundary (FIG. 13). The local area specifying unit is a unit for using the number of subpixels generated with the interpolation process and the size of a local area as its specification data, and a local are an original pixel boundary determining unit is a unit for automatically determining whether or not the local area recognizes original pixels as a boundary.

Provided below are the explanations about the configurations shown in FIGS. 11 through 13.

FIG. 11 exemplifies the third configuration of the locally binarizing unit.

The locally binarizing unit 100 is composed of a pixel point threshold calculating unit 101, an interpolating unit 102, and a comparing unit 83.

When a gray-scale image is input, the data of the original pixels of the gray-scale are input to the subpixel processing unit **80** and the pixel point threshold value calculating unit **101**. The subpixel processing unit **80** generates a predetermined number of subpixels between the original pixels by performing the above described process. The pixel point threshold value calculating unit **101** obtains the threshold value at an original pixel point only from the values of the original pixels. The threshold value is obtained by generating the data shown in FIG. **7**, and using the average value, the variance, etc. The threshold value obtained at each original pixel point is used by an interpolating unit in order to obtain the threshold value of a subpixel with the method such as the linear interpolation method, etc. The comparing unit **83** makes a comparison between the value of an original pixel or the value of a subpixel generated by the subpixel processing unit, and the threshold value at each pixel point, which is obtained by the pixel point threshold value calculating unit **101** and the interpolating unit **102**, and binarizes the pixel value. A binary image is obtained by performing such a process for all of the pixel points.

FIG. **12** is a block diagram exemplifying the fourth configuration of the locally binarizing unit.

A locally binarizing unit **110** is composed of a pixel point threshold value calculating unit **111**, an interpolating unit **117**, and a comparing unit **118**. The contents of the process performed by the subpixel processing unit **80** are similar to those described above. The data of an input gray-scale image is input to the subpixel processing unit **80**, and the pixel point threshold value calculating unit **111** included in the locally binarizing unit **110**. The pixel point threshold value calculating unit **111** obtains the values at the pixel points within the local area centering around a targeted pixel point. The pixel point average value calculating unit **112** calculates an average of the pixel values. A pixel point square average value calculating unit **113** calculates an average of the squares of the pixel values within the local area. The average of the pixel values from the pixel point average value calculating unit **112** and the average of the squares of the pixel values from the pixel point square average value calculating unit **113** are input to a pixel point variance calculating unit **114**, which calculates the variance of the distribution of the pixel values. The calculated variance is input to the pixel point standard deviation value calculating unit **15**, which obtains the standard deviation value of the distribution of the pixel values.

The average, variance, and standard deviation values of the pixel values are input to a linear combination calculating unit **116**, which calculates a threshold value. The expression for calculating a threshold value must be suitably set by a skilled artisan. According to this preferred embodiment, however, the linear combination of the average and standard deviation values is used as described above, and the threshold value is adjusted by multiplying the standard deviation value and a parameter in order to adjust to what degree the standard deviation value is to affect the threshold value to be set.

After the threshold value is obtained for a certain targeted pixel point in this way, this threshold value calculation process is repeatedly performed for the respective original pixel points of an input gray-scale image until the threshold values are obtained for all of the original pixels. These threshold values are transmitted to the interpolating unit **117**, which then obtains the threshold values of subpixels by performing the interpolation process. As a result of the above described processes, the threshold values of the original pixels and the subpixels are obtained. The threshold values are compared with the pixel values transmitted from the subpixel processing unit **80** in the comparing unit **118**, and the pixel values are then binarized and output.

FIG. **13** is a block diagram exemplifying the fifth configuration of the locally binarizing unit.

In this example, a locally binarizing unit **120** is composed of a local area original pixel boundary determining unit **121**, an original pixel boundary pixel point threshold value calculating unit **122**, an original pixel non-boundary pixel point threshold value calculating unit **123**, an interpolating unit **124**, and a comparing unit **125**.

The data of an input gray-scale image are input to the subpixel processing unit **80**, and the above described process is performed. Then, the data after the subpixel generation process is input to the comparing unit **125**.

The data of the input gray-scale image is input also to the local area original pixel boundary determining unit **121** included in the locally binarizing unit **120**. A local area specifying unit **126** is an input unit for specifying where to set a local area for a particular original pixel. The specification may be made either manually or automatically. Especially, if the local area is included by the line structuring a character, the local binarization process may be unsuccessfully performed in some cases. It is therefore desirable that the local area includes the portion of the line structuring a character and the portion of a background in a ratio of 1 to 1.

When the range of the local area is set by the local area specifying unit **126** for the target pixel, the local area original pixel boundary determining unit **121** determines whether or not original pixels exist on the boundary of the local area. If the original pixels exist on the boundary of the local area, data is transmitted to the original pixel boundary pixel point threshold value calculating unit **122**, which is made to calculate the threshold value for each of the original pixels. If the original pixels do not exist on the boundary of the local area, the data is transmitted to the original pixel non-boundary pixel point threshold value calculating unit **123**, which is made to calculate the threshold value for each of the original pixels. The methods for calculating a threshold value, which are executed by the original pixel boundary pixel point threshold value calculating unit **122** and the original pixel non-boundary pixel point threshold value calculating unit **123**, will be described later.

Since the threshold values calculated by the original pixel point threshold value calculating unit **122** or the original pixel non-boundary pixel point threshold value calculating unit **123** are intended only for the original pixels, the threshold values are transmitted to the interpolating unit **124**, which obtains the threshold value for a subpixel by performing the interpolation process.

The threshold values obtained for the original pixels and the subpixels in this way are transmitted to the comparing unit **125**, which makes a comparison between the threshold values and the pixel values transmitted from the subpixel processing unit **80**. The pixels are then binarized and output.

Here, the expressions for directly calculating a local average value and a local square average value from original pixel values will be provided. A local variance and a local standard deviation value are obtained with fundamental arithmetic operations by calculating the local average value and the local square average value, and the linear combination of the local variance and the local standard deviation value is used, whereby a threshold value is obtained according to the above described expressions referred to in the explanation of FIGS. **3A** and **3B**.

FIG. **14** is a schematic diagram explaining the method for calculating a local average value and a local square average value from original pixel values.

First of all, symbols will be defined. A local area centering around an original pixel is illustrated in FIG. 14. The coordinate of the targeted original pixel is assumed to be **(0, 0)** for ease of explanation. The description of the coordinate of a normal original pixel is obtained by parallel translating the coordinate. The degree of the subpixel generation is defined to be $n(\geqq 1)$, and $(n-1)$ new subpixels are inserted between two original pixels. The local area is a square area centering around a targeted original pixel. The upper left point (that is, the coordinate of the upper left original pixel) of the regular square, which is the original pixel farthest from the targeted original point, among the original pixels included in the local area, is assumed to be $(-M, -M)$ with the condition $1 \leqq M$ imposed. The number of subpixels, which exist outside the regular square but within the local area in one direction, after the subpixel generation process is assumed to be "r". "r" must satisfy $0 \leqq r < n$. The original pixel value at a coordinate (i, j) is represented as $I(i, j)$.

Assuming that one side of the local area is "L", the following equation is satisfied.

$$L=2(M^*n+r)+1 \quad (0 \leqq r < n)$$

That is, "M" original pixels center around the targeted pixel point, for example, in a vertical direction, "n−1" subpixels exist between the original pixels, and "r" subpixels exist outside the "M" original pixels. Since the pixels exist in both of the top and the bottom of the local area in a similar manner, $2(M^*n+r)$ is obtained. Additionally, the single target point is added, so that the above described equation is obtained.

"L" is an odd number larger than "2n". Conversely, if "L" which is an odd number larger than a positive number "n", and "2n" are given, "M" and "r" which satisfy the above described equation are uniquely determined. That is, if $(L-1)/2$ is divided by "n" and its quotient and remainder are obtained, the quotient and the remainder respectively correspond to "M" and "r".

Represented below are the equations for obtaining a local average value and a local square average value, which are respectively used in the case (1) where the boundary of a local area is original pixels (r=0), and in a normal case (2). The expressions used in the case (1) are simpler than those used in the case (2). The expressions are represented as follows.

$$\text{local average value} = E(I)$$

$$\text{local square average value} = E(I^2)$$

C_i(i=0, 1, 2, 3, . . .) is a coefficient which will appear in the following expressions. Even if identical symbols are used, their definitions are different depending on the expressions. Since $E(I)$ and $E(I^2)$ are multiplied by C_0 in the following expressions, it is easily understood that the right side of the expressions is divided by c_o in order to obtain $E(I)$ and $E(I^2)$.

(1) In the case where original pixels exist on the boundary of a local area (r=0)

Note that the coefficients are defined by the

$$C_0E(I) = C_1 \sum_{i=-M+1}^{M-1} \sum_{j=-M+1}^{M-1} I(i, j) +$$

$$C_2 \sum_{k=-M+1}^{M-1} \{I(-M, k) + I(M, k) + I(k, -M) + I(k, M)\} +$$

$$C_3\{I(-M, -M) + I(-M, M) + I(M, -M) + I(M, M)\}$$

following equations.

$$C_0=4(2Mn+1)^2$$

$$C_1=4n^2$$

$$C_2=2n(n+1)$$

$$C_3=(n+1)^2$$

$$C_0E(I^2) = \left[C_1 \sum_{i=-M+1}^{M-1} \sum_{j=-M+1}^{M-1} I(i, j)^2 + C_2 \sum_{k=-M+1}^{M-1} \{I(-M, k)^2 + I(M, k)^2 + I(k, -M)^2 + I(k, M)^2\} + C_3\{I(-M, -M)^2 + I(-M, M)^2 + I(M, -M)^2 + I(M, M)^2\}\right] +$$

$$\left[C_4 \sum_{i=-M+1}^{M-1} \sum_{j=-M}^{M-1} I(i, j)I(i, j+1) + C_5 \sum_{j=-M}^{M-1} \{I(-M, j)I(-M, J+1) + I(M, j)I(M, j+1)\}\right] +$$

$$\left[C_4 \sum_{i=-M}^{(} M-1) \sum_{j=-M+1}^{M-1} I(i, j)I(i+1, j) + C_5 \sum_{i=-M}^{M-1} \{I(i, -M)I(i+1, -M) + I(i, M)I(i+1, M)\}\right] +$$

$$\left[C_6 \sum_{i=-M}^{M-1} \sum_{j=-M}^{M-1} \{I(i, j)I(i+1, j+1) + I(i, j+1)I(i+1, j)\}\right]$$

Note that the coefficients are defined by the following equations.

(2) In a normal case

$$C_0=36(2Mn+1)^2n^2$$

$$C_1=4(2n^2+1)^2$$

$$C_2=2(n+1)(2n+1)(2n^2+1)$$

$$C_3=(n+1)^2(2n+1)^2$$

$$C_4=4(n^2-1)(2n^2+1)$$

$$C_5=2(n+1)(n^2-1)(2n+1)$$

$$C_6=2(n^2-1)^2$$

$$C_0E(I) = C_1 \sum_{i=-M+1}^{M-1} \sum_{j=-M+1}^{M-1} I(i, j) + C_2 \sum_{k=-M+1}^{M-1} \{I(-M, k) + I(M, k) + I(k, -M) + I(k, M)\} + C_3\{I(-M, -M) + I(-M, M) + I(M, -M) + I(M, M)\} +$$

$$C_4 \sum_{k=-M+1}^{M-1} \{I(-M-1, k) + I(M+1, k) + I(k, -M-1) + I(k, M+1)\} + C_5\{I(-M-1, -M) + I(-M-1, M) + I(M+1, -M) + I(M+1, M) + I(-M, -M-1) + I(-M, M+1) + I(M, -M-1) + I(M, M+1)\} +$$

$$C_6\{I(-M-1, -M, -1) + I(-M-1, M+1) + I(M+1, -M-1) + I(M+1, M +$$

Note that the coefficients are defined by the following equations.

$$C_0=4(2Mn+1)^2n^2$$

$$C_1=4n^4$$

$$C_2=2n^2\{n^2+(2r+1)n-r(r+1)\}$$

$$C_3=n^4+2(2r+1)n^3+(2r^2+2r+1)n^2-2r(r+1)(2r+1)n+r^2(r+1$$

$$C_4=2r(r+1)n^2$$

$$C_5=r(r+1)\{n^2+(2r+1)n-r(r+1)\}$$

$$C_6=r^2(r+1)^2$$

Note that the coefficients are defined by the following equations.

$$
\begin{aligned}
C_0E(I^2) = \Bigg[& C_1\sum_{i=-M+1}^{M-1}\sum_{j=-M+1}^{M-1} I(i,j)^2 + C_2\sum_{k=-M+1}^{M-1}\{I(-M,k)^2 + \\
& I(M,k)^2 + I(k,-M)^2 + I(k,M)^2\} + C_3\{I(-M,-M)^2 + \\
& I(-M,M)^2 + I(M,-M)^2 + I(M,M)^2\} + \\
& C_4\sum_{k=-M+1}^{M-1}\{I(-M-1,k)^2 + I(M+1,k)^2 + \\
& I(k,-M-1)^2 + I(k,M+1)^2\} + C_5\{I(-M-1,-M)^2 + \\
& I(-M-1,M)^2 + I(M+1,-M)^2 + I(M+1,M)^2 + \\
& I(-M,-M-1)^2 + I(-M,M+1)^2 + I(M,-M-1)^2 + \\
& I(M,M+1)^2\} + C_6\{I(-M-1,-M-1)^2 + \\
& I(-M-1,M+1)^2 + I(M+1,-M-1)^2 + I(M+1,M+ \\
& \Bigg[C_7\sum_{i=-M+1}^{M-1}\sum_{j=-m}^{M-1} I(i,j)I(i,j+1) + \\
& C_8\sum_{j=-M}^{M-1}\{I(-M,j)I(-M,j+1) + I(M,j)I(M,j+1)\} + \\
& C_9\sum_{j=-M}^{M-1}\{I(-M-1,j)I(-M-1,j+1) + \\
& I(M+1,j)I(M+1,j+1)\} + \\
& C_{10}\sum_{i=-M+1}^{M-1}\{I(i,-M-1)I(j,-M) + I(i,M)I(i,M+1)\} + \\
& C_{11}\{I(-M,-M-1)I(-M,-M) + \\
& I(-M,M)I(-M,M+1) + I(M,-M-1)I(M,-M) + \\
& I(M,M)I(M,M+1)\} + \\
& C_{12}\{I(-M-1,-M-1)I(-M-1,-M) + \\
& I(-M-1,M)I(-M-1,M+1) + \\
& I(M+1,-M-1)I(M+1,-M) + \\
& I(M+1,M)I(M+1,M+1)\}\Bigg] + \\
& \Bigg[C_7\sum_{i=-M+1}^{M-1}\sum_{j=-M+1}^{M-1} I(i,j)I(i+1,j) + \\
& C_8\sum_{i=-M}^{M-1}\{I8i,-M\Big)I(i+1,-M) + I(i,M)I(i+1,M)\} +
\end{aligned}
$$

-continued

$$
\begin{aligned}
& C_9\sum_{j=-M}^{M-1}\{I(i,-M-1)I(i+1,-M-1) + \\
& I(i,M+1)I(i+1,M+1)\} + \\
& C_{10}\sum_{j=-M+1}^{M-1}\{I(-M-1,j)I(-M,j) + \\
& I(M,j)I(M+1,j)\} + C_{11}\{I(-M-1,-M)I(-M,-M) + \\
& I(M,-M)I(M+1,-M) + I(-M-1,M)I(-M,M) + \\
& I(M,M)I(M+1,M)\} + \\
& C_{12}\{I(-M-1,-M-1)I(-M,-M-1) + \\
& I(M,-M-1)I(M+1,-M-1) + \\
& I(-M-1,M+1)I(-M,M+1) + \\
& I(M,M+1)I(M+1,M+1)\}\Bigg] + \\
& \Bigg[C_{13}\sum_{k=-M}^{M-1}\sum_{j=-M}^{M-1}\{I(i,j)I(i+1,j+1) + \\
& I(i,j+1)I(i+1,j)\} + \\
& C_{14}\sum_{k=-M}^{M-1}(\{I(-M-1,k)I(-M,k+1) + \\
& I(-M-1,k+1)I(-M,k)\} + \{I(M,k)I(M+1,k+1) + \\
& I(M,k+1)I(M+1,k)\} + \{I(k,-M-1)I(k+1,-M) + \\
& I(k+1,-M-1)I(k,-M)\} + \\
& \{I(k,M)I(k+1,M+1)LI(k+1,M)I(-M,-M-1)\}) + \\
& C_{15}(\{I(-M-1,-M-1)I(-M,-M) + \\
& I(-M-1,-M)I(-M,-M-1)\} + \\
& \{I(-M-1,M)I(-M,M+1) + \\
& I(-M-1,M+1)I(-M,M)\} + \\
& I(M,-M-1)I(M+1,-M) + \\
& I(M,-M)I(M+1,-M-1)\} + \\
& \{I(M,M)I(M+1,M+1) + I(M,M+1)I(M+1,M)\})
\end{aligned}
$$

The expression which is used in the case where "M" is equal to or larger than "1" is represented for the

$$C_0-36n^4(2MN+2r+1)^2$$

$$C_1=4n^2(2n^2+1)^2$$

$$C_2=2n\{4n^5+6(2r+1)n^44(-3r^2-3r+1)n^3+(4r^3+6r^2+8r+3)n^2+(-6r^2-6r+1)n+r(r+1)(2r+1)\}$$

$$C_3=4n^6+12(2r+1)n^5+(12r^2+12r+13)n^4-2(32r^3+48r^2+10r-3)n^3$$

$$+(60r^4+120r^3+54r^2-6r+1)n^2-2r(12r^4+30r^3+22r^2+3r-1)n+r^2(r+1)^2(2r+1)^2$$

$$C_4=2r(r+1)(2r+1)n(2n^2+1)$$

$$C_5=r(r+1)(2r+1)\{2n^3+3(2r+1)n^2+(-6r^2-6r+1)n+r(r+1)(2r1)\}$$

$$C_6=r^2(r+1)^2(2r+1)^2$$

$$C_74n^2(n^2-1)(2n^1+1)$$

$$C_82n(n^2-1)\{2n^3+3(2r+1)n^2+(-6r^2-6r+1)n+r(r+1)(2r+1)$$

$$C_9=2r(r+1)(2r+1)n(n^2-1)$$

$$C_{10}=4r(r+1)n(3n-(2r+1))(2n^2+1)$$

$$C_{11}=2r(r+1)\{6n^4+7(2r+1)n^3-30r(r+1)n^2+(18r^3+27r^2+7r-1)n-r(4r^3+8r^2+5r+1)\}$$

$$C_{12}=2r^2(r+1)^2(2r+1)(3n-(2r+1))$$

$$C_{13}=2n^2(n^2-1)^2$$

$$C_{14}=2r(r+1)n(n^2-1)(3n-(2r+1))$$

$$C_{15}=2r^2(r+1)^2(3n-(2r+1))$$

normal case (2). The expression which is used in the case where M=0 can be obtained in the following three steps.

(1) Setting "M" to "1".

(2) Replacing "n" with "r".

(3) Replacing the value of "I" with the boundary value of a local area

The variance and standard deviation values can be obtained by using the average value and the square average value of the pixel values within a local area, which are obtained with the above described expressions, thereby obtaining the expression for determining a threshold value.

FIG. 15 is a block diagram exemplifying the sixth configuration of the locally binarizing unit.

With this configuration, the value obtained by interpolating the difference between the pixel value at an original pixel point of a gray-scale image and the local threshold value after the subpixel generation process, at an interpolation point (subpixel), and the value of the binary image at the interpolation point, are determined with the sign of the interpolation value. Because the interpolation process is reduced from twice to once with this configuration, the processing speed can be improved. Especially, if the subpixel generation process is performed with the linear interpolation method, the same effects as those in the case where respective original pixels are interpolated and compared can be obtained.

the subpixel value for {original pixel value−binarization threshold value}=the value obtained by performing the subpixel generation process for original pixel values−the value obtained by performing the subpixel generation process for a binarization threshold value

That is, the locally binarizing unit 140 is composed of a pixel point threshold value calculating unit 140, a difference calculating unit 142, an interpolating unit 143, and a sign determining unit 144.

The data of an input gray-scale image are directly input to the difference calculating unit 142, and to the pixel point threshold value calculating unit 141. The pixel point threshold value calculating unit 141 calculates the threshold value within a local area only from the original pixel values included in the local area for the original pixel point of the input gray-scale image, and generates the threshold value for a targeted original pixel. Next, the targeted original pixel value and the generated threshold value are input to the difference calculating unit 142, which calculates the difference between them. The value of a subpixel is then obtained by interpolating this difference in a similar manner as in the above described process for interpolating the level of brightness (gray scale). Since the obtained value is the same as the value of the difference between the value of the subpixel and the threshold value, the difference from the value of the original pixel and that of the subpixel are input to the sign determining unit 144, which examines the sign of the difference.

For example, if the pixel value is larger than the threshold value, the above described difference becomes positive, and, for example, the value "0" is assigned to this pixel when being binarized. If the pixel value is smaller than the threshold value, the difference becomes negative, and, for example, the value "1" is assigned to the pixel when being binarized. By performing such a process for all of the pixel points, a binary image can be obtained.

FIG. 16 is a block diagram exemplifying the seventh configuration of the local binarizing unit.

With this configuration, a binary image is obtained by obtaining the local threshold value after the subpixel generation process at each pixel point of a gray-scale image, and by using the table to which the values at 4 pixel points of the gray-scale image and the local threshold value and from which the binary image enclosed by 4 pixel points is output as data.

Namely, a locally binarizing unit 150 is composed of a pixel point threshold value calculating unit 151, a binary image searching unit 152, and a memory 153. The pixel point threshold value calculating unit 151 obtains the threshold value in a local area from the pixel values of the input gray-scale image for each of the pixels, and provides it to the binary image searching unit 152. The binary image searching unit 152 receives the threshold values from the pixel point threshold value calculating unit 151 and the original pixel values of the gray-scale image, and selects 4 original pixels forming the regular square which is a minimum unit of the grid formed by original pixel points. Then, a binary image is obtained by referencing the table stored in the memory 153 based on the pixel values and the threshold values of the 4 original pixels. The table stored in the memory 153 is a table to which the binary image data is registered for the combination of the pixel values and the threshold values of 4 original pixels. The binary image searching unit 152 obtains the binary image data for all of unit grids (minimum regular squares forming a grid) structured by the original pixels of the input gray-scale image according to this table, generates an entire binary image by combining the data, and outputs the generated image.

Only the explanations about the configurations of the variable resolution binarizing unit were provided above. However, as shown in FIG. 6, the rough extraction process for roughly extracting a drawing area is initially performed for a gray-scale image, and the above described process is performed only for the extracted drawing area, thereby reducing the amount of processing time.

FIG. 17 shows the processing examples up to the process for binarizing a color image, and FIG. 18 shows a gray-scale image according to this preferred embodiment.

FIG. 17 shows the processing example of a 150-dpi color image.

The top image is a color image document with a resolution of 150 dpi. In an actual color image, various colors appear around a black color representing characters. The top color image converted into a gray-scale image is shown as a middle image. The gray-scale image binarized with the conventional method is the binary image on the right at the bottom. Viewing this image, the detailed portions of the characters are defaced and the characters are difficult to recognize. In the meantime, the binary image obtained by performing the processing according to this preferred embodiment is the image on the left at the bottom. Since subpixels are generated according to this preferred embodiment, the resolution of the binary image is substantially higher than that of the original color image. In this case, the amount of information increases. However, since the binary image data is used not for printing by a printer, etc, but for charter recognition as it is, the clearer representation of the characters facilitates the character recognition.

FIG. 18 shows the processing example of 150- and 100-dpi gray-scale images.

If the 150- and 100-dpi gray-scale images are binarized by a conventional process, an erroneous character recognition result may often be obtained due to the defacement of the characters. In the meantime, according to this preferred embodiment, subpixels are generated and the information about the levels of brightness can be prevented from being lost, whereby the characters can be recognized more clearly, and a higher recognition rate can be realized when the character recognition process is performed.

FIG. 19 is a block diagram explaining the configuration of the hardware required for implementing this preferred embodiment as software.

In this preferred embodiment, the subpixel generation process, the interpolation process, the threshold value calculation process, etc. can be implemented as a program running on a computer. In this case, as the hardware configuration required for the computer, a CPU **181** for performing the above described processes, a RAM **183** for storing the program for performing these processes in an executable form, etc. must be interconnected by a bus **180**, so that they can communicate with each other. Furthermore, a ROM **182** for storing the BIOS required for running the CPU **181**, a storage device **189** for storing the program, etc. are arranged. The storage device **189** is implemented, for example, as a hard disk, etc. In addition, a storage medium reading device **187** is required when the program is stored onto a portable storage medium **188** such as a floppy disk, a CD-ROM, etc. and used. The program read from the storage device **189** or the portable storage medium **188** is expanded and stored in the RAM **183** so that the CPU **181** can execute it. Furthermore, an input/output device **186** composed of a monitor, a keyboard, a mouse, etc. is arranged in order to transmit to the CPU **181** the commands issued by a user who operates the device, and to display the results of the processes performed by the CPU **181** for the user.

Furthermore, the program may not be stored in a computer used by a user. It may be used depending on need by being downloaded from a database possessed by a program provider **185**. Or, the program may be executed in a network by connecting the user and the program provider **185** via a LAN. Only a command input and a result display are performed by the computer possessed by the user in this case.

As described above in detail, according to the present invention, a color or gray-scale document image can be quickly binarized with high accuracy, thereby recognizing the image accurately and rapidly.

What is claimed is:

1. A document image recognizing device, comprising:

image converting means for converting an input document image into a gray-scale image if the input document image is a color image, and for newly outputting a gray-scale image if the input image is a gray-scale image;

variable resolution binarizing means for converting the input document image into a binary image having a higher resolution according to a resolution of the gray-scale image; wherein said variable resolution binarizing means performs a sub-pixel generation process for increasing a number of pixels included in an image by interpolating pixel values ofa gray-scale image, sets a local threshold value within a local area centering around a particular pixel, and obtains a binary image by using the local threshold value; and

recognizing means for recognizing the binarized image.

2. The device according to claim 1, wherein:

said recognizing means recognizes a converted or input binary image, and converts the binary image into electronic codes.

3. The device according to claim 1, further comprising:

drawing area roughly extracting means for roughly extracting a drawing area according to a global threshold value for pixel values of a gray-scale image, wherein

said recognizing means recognizes a binary image in an area extracted by said drawing area roughly extracting means.

4. The device according to claim 3, wherein:

the global threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

5. The device according to claim 3, wherein

said variable resolution binarizing means performs the subpixel generation process for an entire input gray-scale image; and

said drawing area roughly extracting means roughly extracts a drawing area from gray-scale image data for which the subpixel generation process is performed.

6. The device according to claim 1, wherein:

the subpixel generation process is performed by linearly interpolating original pixel values of the gray-scale image.

7. The device according to claim 1, wherein:

said variable resolution binarizing means obtains a binary image by binarizing pixel values by using the local threshold value obtained from a distribution of the pixel values within a local area including pixels; and

the local threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

8. The device according to claim 1, further comprising:

drawing area roughly extracting means for performing a global process which roughly extracts a drawing area according to a global threshold value for pixel values of a gray-scale image, wherein

said variable resolution binarizing means performs the subpixel generation process for increasing a number of pixels included in an image by interpolating pixel values of a gray-scale image, for a drawing area which is roughly extracted with the global process, and performs a local binarization process by using the local threshold value for each pixel included in the roughly extracted drawing area.

9. The device according to claim 1, wherein the local threshold value at a pixel point after the subpixel generation process is based on obtaining a value of a subpixel generated with the subpixel generation process from original pixel values, and obtaining a local threshold value from subpixel values.

10. The device according to claim 1, wherein a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image, and a local threshold value of a subpixel is obtained by interpolating local threshold values at pixel points of the gray-scale image.

11. The device according to claim 10, wherein the local threshold value at the pixel point of the gray-scale image is obtained by using a value of a subpixel obtained by performing the subpixel generation process for pixel points of the gray-scale image.

12. The device according to claim 10, wherein the local threshold value at the pixel point of the gray-scale image after the subpixel generation process is based on obtaining a subpixel value from original pixel values, and obtaining a local threshold value from subpixel values.

13. The device according to claim 12, wherein the local threshold value is obtained, using subpixel values, from a local threshold value from a local area that recognizes original pixels as its boundary.

14. The device according to claim 10, wherein an interpolation value at a subpixel point is obtained by interpolating a difference between a value at a pixel point of a gray-scale image and a local threshold value after the subpixel generation process, and a value of a binary image at the subpixel point is determined with a sign of the interpolation value.

15. The device according to claim 1, wherein a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

16. The device according to claim 1, wherein:

the local area recognizes original pixels as its boundary; and

a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

17. The device according to claim 1, further comprising:

specifying means for specifying a range of a local area, wherein

whether or not the local area recognizes original pixels as its boundary is determined by using as specification data a number of subpixels generated with the subpixel generation process and a size of the local area.

18. The device according to claim 1, wherein:

a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image;

a table to which pixel values and local threshold values at four pixel points of a gray-scale are input and from which a binary image enclosed by the four pixel points is output as data, is included; and

a binary image is obtained by using said table.

19. A document image recognizing method, comprising:

converting an input document image into a gray-scale image if the input document image is a color image, and newly outputting a gray-scale image if the input image is a gray-scale image;

converting the input document image into a binary image having a higher resolution according to a resolution of the gray-scale image; wherein said converting the input document image into a binary image performs a subpixel generation process for increasing a number of pixels included in an image by interpolating pixel values of a gray-scale image, sets a local threshold value within a local area centering around a particular pixel, and obtains a binary image by using the local threshold value; and

recognizing the binary image.

20. The method according to claim 19, wherein:

said recognizing the binary image recognizes a converted or input binary image, and converts the binary image into electronic codes.

21. The method according to claim 20, further comprising:

roughly extracting a drawing area according to a global threshold value for pixel values of a gray-scale image, wherein

said recognizing the binary image recognizes a binary image in an area extracted by said roughly extracting a drawing area.

22. The method according to claim 21, wherein the global threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

23. The method according to claim 21, wherein:

said converting the input document image into a binary image performs the subpixel generation process for an entire input gray-scale image; and

said roughly extracting a drawing area roughly extracts a drawing area from gray-scale image data for which the subpixel generation process is performed.

24. The method according to claim 19, wherein:

the subpixel generation process is performed by linearly interpolating original pixel values of the gray-scale image.

25. The method according to claim 19, wherein:

said converting the input document image into a binary image obtains a binary image by binarizing pixel values by using the local threshold value obtained from a distribution of the pixel values within a local area including pixels; and

the local threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

26. The method according to claim 19, further comprising:

performing a global process which roughly extracts a drawing area according to a global threshold value for pixel values of a gray-scale image, wherein

said converting the input document image into a binary image performs the subpixel generation process for increasing a number of pixels included in an image by interpolating pixel values of a gray-scale image, for a drawing area which is roughly extracted with the global process, and performs a local binarization process by using the local threshold value for each pixel included in the roughly extracted drawing area.

27. The method according to claim 19, wherein the local threshold value at a pixel point after the subpixel generation process is based on obtaining a value of a subpixel generated with the subpixel generation process from original pixel values, and obtaining a local threshold value from subpixel values.

28. The method according to claim 19, wherein a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image, and a local threshold value of a subpixel is obtained by interpolating local threshold values at pixel points of the gray-scale image.

29. The method according to claim 28, wherein the local threshold value at the pixel point of the gray-scale image after the subpixel generation process is based on obtaining a subpixel value from original pixel values, and obtaining a local threshold value from subpixel values.

30. The method according to claim 29, wherein the local threshold value is obtained, using subpixel values, from a local threshold value within a local area that recognizes original pixels as its boundary.

31. The method according to claim 28, wherein an interpolation value at a subpixel point is obtained by interpolating a difference between a value at a pixel point of a gray-scale image and a local threshold value after the subpixel generation process, and a value of a binary image at the subpixel point is determined by a sign of the interpolation value.

32. The method according to claim 19, wherein the local threshold value at the pixel point of the gray-scale image is obtained by using a value of a subpixel obtained by performing the subpixel generation process for pixel points of the gray-scale image.

33. The method according to claim 19, wherein a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

34. The method according to claim 19, wherein:

the local area recognizes original pixels as its boundary; and

a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

35. The method according to claim 19, further comprising:

specifying a range of a local area, wherein whether or not the local area recognizes original pixels as its boundary is determined by using as specification data a number of subpixels generated with the subpixel generation process and a size of the local area.

36. The method according to claim 19, wherein:

a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image;

a table to which pixel values and local threshold values at four pixel points of a gray-scale are input and from which a binary image enclosed by the four pixel points is output as data, is included; and

a binary image is obtained by using said table.

37. A computer-readable storage medium for directing a computer to execute a process comprising:

converting an input document image into a gray-scale image if the input document image is a color image, and newly outputting a gray-scale image if the input image is a gray-scale image;

converting the input document image into a binary image having a higher resolution according to a resolution of a gray-scale image; wherein said converting the input document image into a binary image performs a subpixel generation process for increasing a number of pixels included in an image by interpolating pixel values of a gray-scale image, sets a local threshold value within a local area centering around a particular pixel, and obtains a binary image by using the local threshold value; and

recognizing the binary image.

38. The storage medium according to claim 37, wherein:

said recognizing the binary image recognizes a converted or input binary image, and converts the binary image into electronic codes.

39. The storage medium according to claim 37, wherein the process further comprises:

roughly extracting a drawing area according to a global threshold value for pixel values of a gray-scale image, and wherein

said recognizing the binary image recognizes a binary image in an area extracted by said roughly extracting a drawing area.

40. The storage medium according to claim 39, wherein the global threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

41. The storage medium according to claim 39, wherein:

said converting the input document image into a binary image performs the subpixel generation process for an entire input gray-scale image; and

said roughly extracting a drawing area roughly extracts a drawing area from gray-scale image data for which the subpixel generation process is performed.

42. The storage medium according to claim 39, wherein the subpixel generation process is performed by linearly interpolating original pixel values of the gray-scale image.

43. The storage medium according to claim 37, wherein:

said converting the input document image into a binary image obtains a binary image by binarizing pixel values by using the local threshold value obtained from a distribution of the pixel values within a local area including pixels; and

the local threshold value uses a linear combination of an average pixel value, a standard deviation value, and a variance.

44. The storage medium according to claim 37, further comprising:

performing a global process which roughly extracts a drawing area according to a global threshold value for pixel values of a gray-scale image, and wherein

said converting the input document image into a binary image performs the subpixel generation process for increasing a number of pixels included in an image by interpolating pixel values of a gray-scale image, for a drawing area which is roughly extracted with the global process, and performs a local binarization process by using the local threshold value for each pixel included in the roughly extracted drawing area.

45. The storage medium according to claim 37, wherein the local threshold value at a pixel point after the subpixel generation process is based on obtaining a value of a subpixel generated with the subpixel generation process

from original pixel values, and obtaining a local threshold value from subpixel values.

46. The storage medium according to claim **37**, wherein a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image, and a local threshold value of a subpixel is obtained by interpolating local threshold values at pixel points of the gray-scale image.

47. The storage medium according to claim **46**, wherein the local threshold value at the pixel point of the gray-scale image is obtained by using a value of a subpixel obtained by performing the subpixel generation process for pixel points of the gray-scale image.

48. The storage medium according to claim **46**, wherein the local threshold value at the pixel point of the gray-scale image after the subpixel generation process is based on obtaining a subpixel value from original pixel values, and obtaining a local threshold value from subpixel values.

49. The storage medium according to claim **48**, wherein the local threshold value is obtained, using subpixel values, from a local threshold value within a local area that recognizes original pixels as its boundary.

50. The storage medium according to claim **46**, wherein an interpolation value at a subpixel point is obtained by interpolating a difference between a value at a pixel point of a gray-scale image and a local threshold value after the subpixel generation process, and a value of a binary image at the subpixel point is determined with a sign of the interpolation value.

51. The storage medium according to claim **37**, wherein a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

52. The storage medium according to claim **37**, wherein:

the local area recognizes original pixels as its boundary; and

a local threshold value at an original pixel point of a gray-scale image after the subpixel generation process is obtained from a linear combination of an average value, a standard deviation value, and a variance, by obtaining the standard deviation value and the variance after obtaining the average value and a square average value of values at pixel points of the gray-scale image, for which the subpixel generation process is performed, based on obtaining a subpixel value from original pixel values, and obtaining the average value and the square average value from subpixel values.

53. The storage medium according to claim **37**, further comprising:

specifying a range of a local area, and wherein

whether or not the local area recognizes original pixels as its boundary is determined by using as specification data a number of subpixels generated with the subpixel generation process and a size of the local area.

54. The storage medium according to claim **37**, wherein:

a local threshold value after the subpixel generation process is obtained at a pixel point of a gray-scale image;

a table to which pixel values and local threshold values at four pixel points of a gray-scale are input and from which a binary image enclosed by the four pixel points is output as data, is included; and

a binary image is obtained by using said table.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 6,347,156 B1
DATED : February 12, 2002
INVENTOR(S) : Hiroshi Kamada et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 21,
Line 60, after "image;" insert -- and recognizing means for recognizing the binarized image, --
Line 63, change "ofa" to -- of a --.
Line 66, change "; and" to -- . --
Line 67, delete the line.

Column 23,
Line 66, after "image;" insert -- and recognizing the binary image --

Column 24,
Line 3, change "," to -- . -- and delete the rest of the line.
Lines 4-7, delete the lines in their entirety.

Column 26,
Line 5, after "image" insert -- ; and recognizing the binary image, -- and delete the rest of the line.
Line 12, change "; and" to -- . --
Line 13, delete the line.

Signed and Sealed this

Thirteenth Day of August, 2002

Attest:

JAMES E. ROGAN

Attesting Officer *Director of the United States Patent and Trademark Office*

103 cl. 7

US006366696B1

(12) **United States Patent**
Hertz et al.

(10) **Patent No.:** **US 6,366,696 B1**
(45) **Date of Patent:** **Apr. 2, 2002**

(54) **VISUAL BAR CODE RECOGNITION METHOD**

(75) Inventors: **Lois H. Hertz**, Atlanta; **John C. Ming**, Acworth, both of GA (US); **Kevin K. Su**, Petaluma, CA (US)

(73) Assignee: **NCR Corporation**, Dayton, OH (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **08/777,672**

(22) Filed: **Dec. 20, 1996**

(51) **Int. Cl.**^{7} **G06K 9/00**
(52) **U.S. Cl.** **382/183**; 382/181
(58) **Field of Search** 382/182, 183, 382/309, 181, 295, 312; 235/462, 470

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,744,025	A	*	7/1973	Bilgutay	382/309
3,854,035	A		12/1974	Tyler et al.	235/61.11 E
4,027,142	A	*	5/1977	Paup et al.	235/379
4,044,227	A		8/1977	Holm et al.	235/61.7 R
4,147,295	A	*	4/1979	Nojiri et al.	235/470
4,192,452	A		3/1980	Hashimoto et al.	235/471
4,408,344	A	*	10/1983	McWaters et al.	382/309
4,473,746	A		9/1984	Edmonds	250/216
4,641,347	A	*	2/1987	Clark et al.	382/309
4,680,457	A		7/1987	Robertson	235/470
4,748,317	A	*	5/1988	Satoh	235/462
4,806,741	A		2/1989	Robertson	235/462
4,902,883	A		2/1990	Poland	235/462
4,939,355	A		7/1990	Rando et al.	235/467
4,958,064	A		9/1990	Kirkpatrick	235/384
5,086,215	A	*	2/1992	Carsner et al.	235/462
5,115,888	A		5/1992	Schneider	186/61
5,120,940	A		6/1992	Willsie	235/462
5,194,722	A	*	3/1993	Mergenthaler et al.	235/470
5,247,154	A		9/1993	Ahmed	219/121.83
5,276,315	A	*	1/1994	Surka	235/462.1
5,352,878	A	*	10/1994	Smith et al.	235/462
5,367,578	A	*	11/1994	Golem et al.	382/183
5,463,700	A	*	10/1995	Nakazawa	382/309
5,587,575	A	*	12/1996	Leitner et al.	235/462

* cited by examiner

Primary Examiner—Yon J. Couso
(74) *Attorney, Agent, or Firm*—Paul W. Martin; Priest & Goldstein, PLLC

(57) **ABSTRACT**

A visual bar code recognition method which combines conventional decoding techniques with optical character recognition (OCR). The visual bar code recognition method captures an image of an object containing a bar code. Regardless of the orientation of the bar code within the field-of-view, the system detects the presence of the bar code, and decodes it using the bar/space patterns. It then produces an independent decoding of the human-readable numbers printed on the bar code using OCR. From these two decodings, it determines the identity of the object. It verifies this identity by comparing the physical characteristics of the object from the image with the known features of the product.

14 Claims, 5 Drawing Sheets

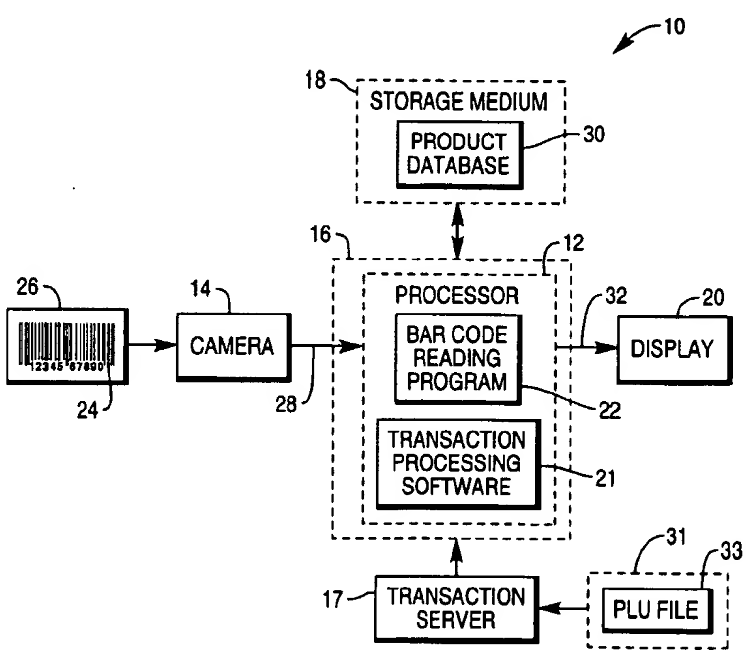


FIG. 1

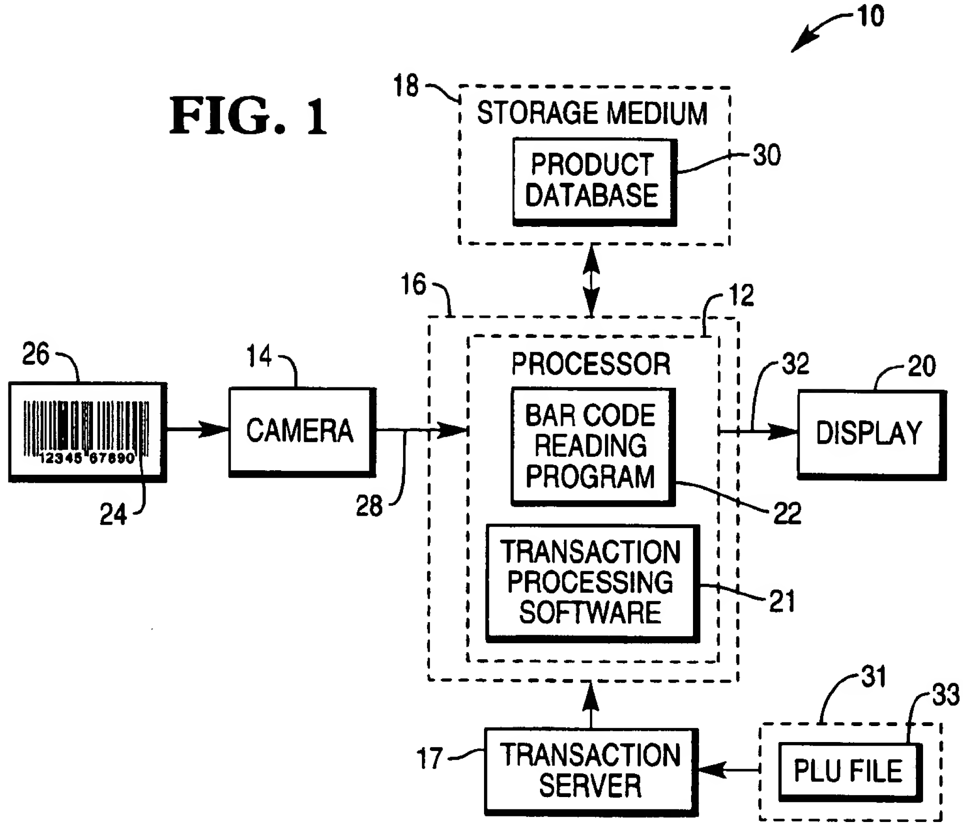

10
18
STORAGE MEDIUM
PRODUCT DATABASE
30
16
12
PROCESSOR
BAR CODE READING PROGRAM
22
TRANSACTION PROCESSING SOFTWARE
21
26
12345 67890
24
14
CAMERA
28
32
20
DISPLAY
17
TRANSACTION SERVER
31
33
PLU FILE

FIG. 2

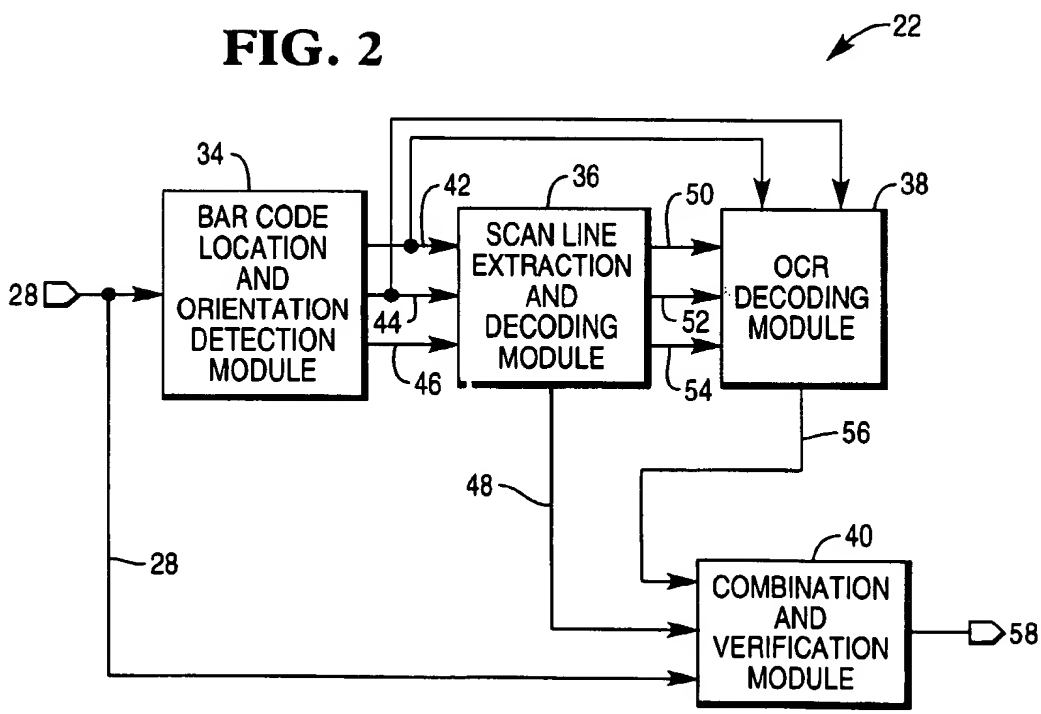

22
34
BAR CODE LOCATION AND ORIENTATION DETECTION MODULE
28
42
44
46
36
SCAN LINE EXTRACTION AND DECODING MODULE
50
52
54
38
OCR DECODING MODULE
56
48
40
COMBINATION AND VERIFICATION MODULE
58
28

FIG. 3

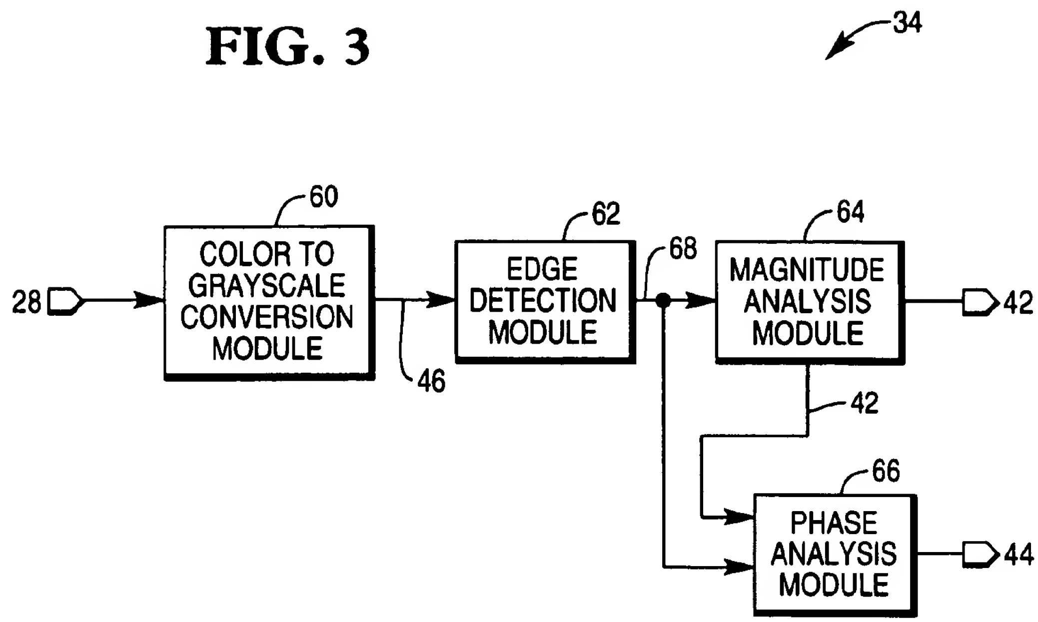

34
60
COLOR TO GRAYSCALE CONVERSION MODULE
62
EDGE DETECTION MODULE
68
64
MAGNITUDE ANALYSIS MODULE
28
46
42
42
66
PHASE ANALYSIS MODULE
44

FIG. 4

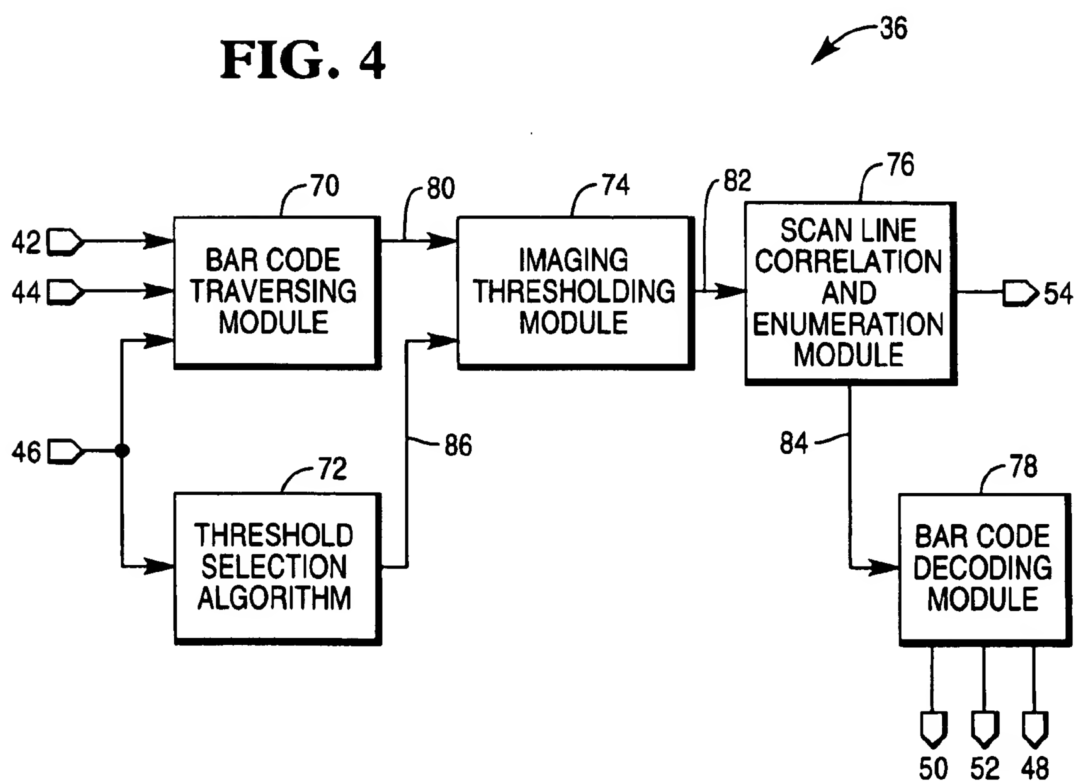

36
70
BAR CODE TRAVERSING MODULE
80
74
IMAGING THRESHOLDING MODULE
82
76
SCAN LINE CORRELATION AND ENUMERATION MODULE
42
44
46
54
72
THRESHOLD SELECTION ALGORITHM
86
84
78
BAR CODE DECODING MODULE
50
52
48

FIG. 5

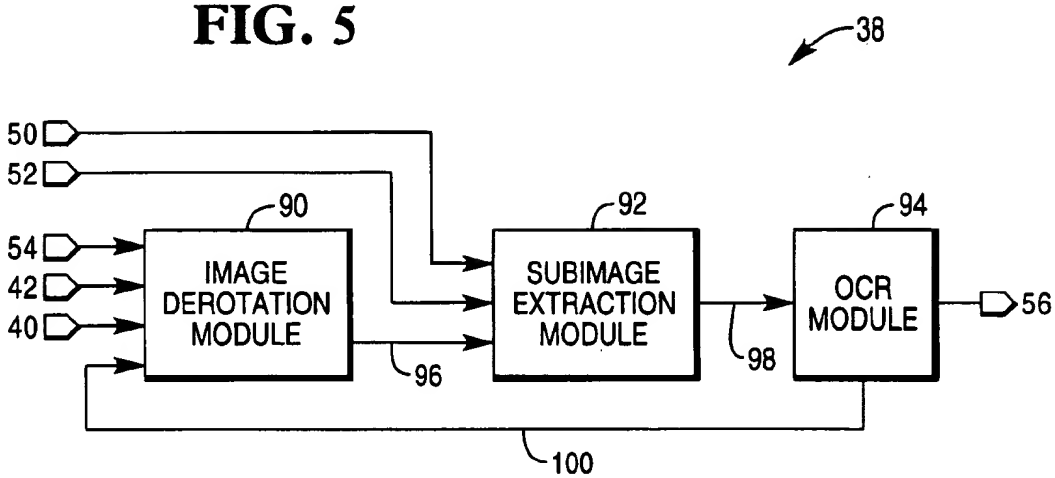

38
50
52
54
42
40
90
IMAGE DEROTATION MODULE
92
SUBIMAGE EXTRACTION MODULE
94
OCR MODULE
56
96
98
100

FIG. 6

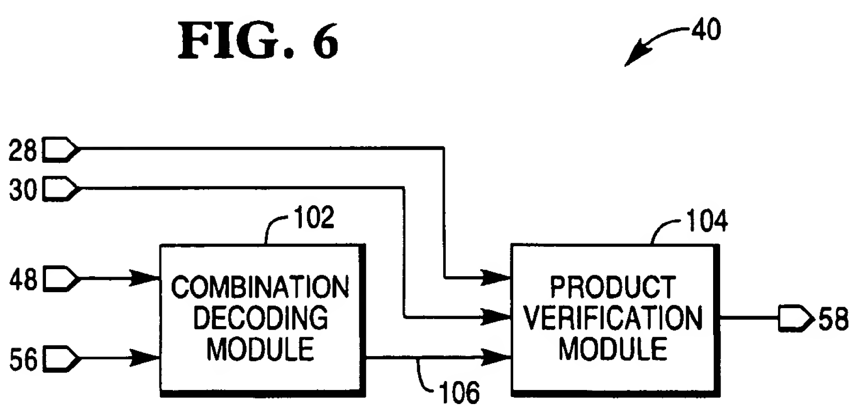

40
28
30
48
56
102
COMBINATION DECODING MODULE
104
PRODUCT VERIFICATION MODULE
58
106

FIG. 7A

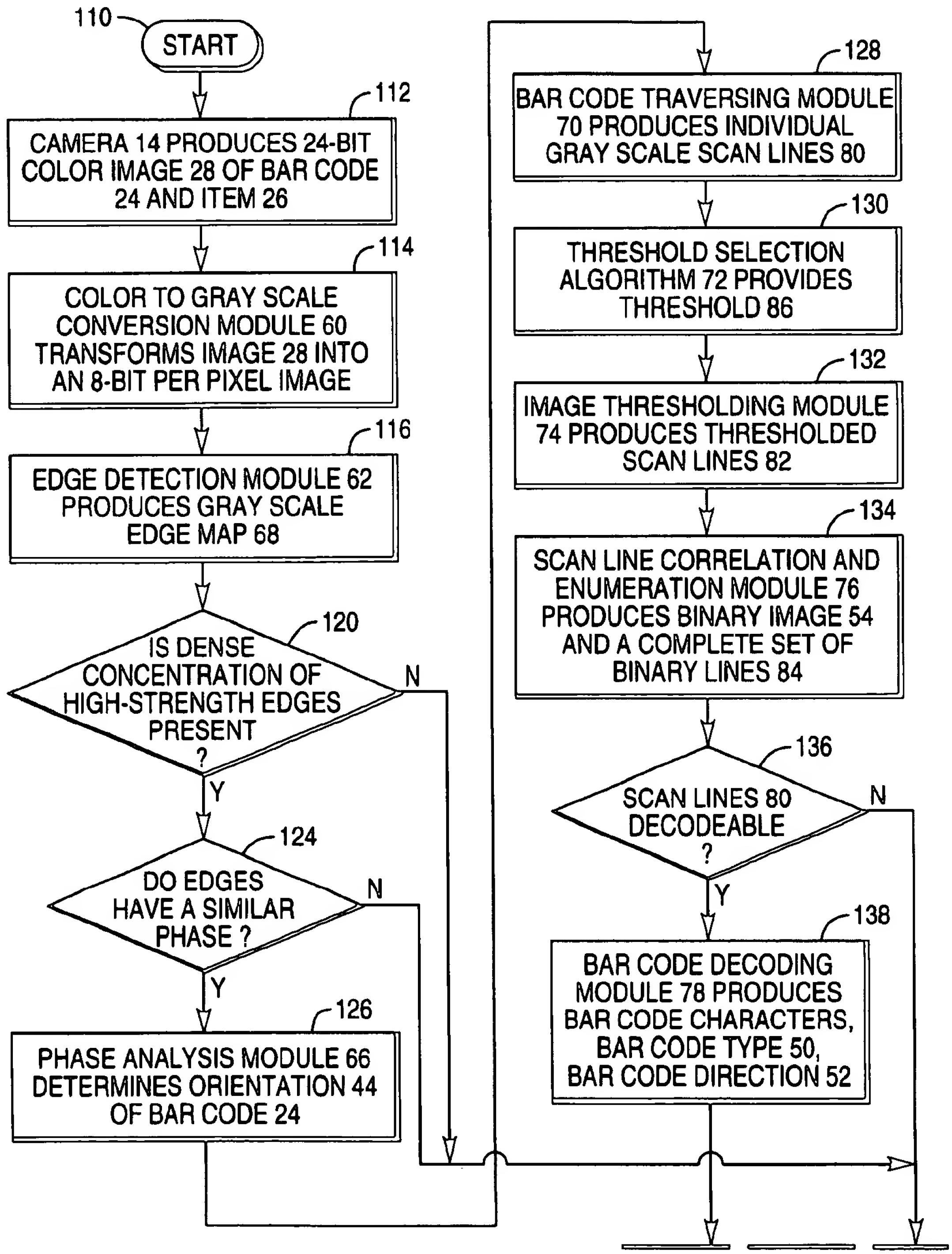
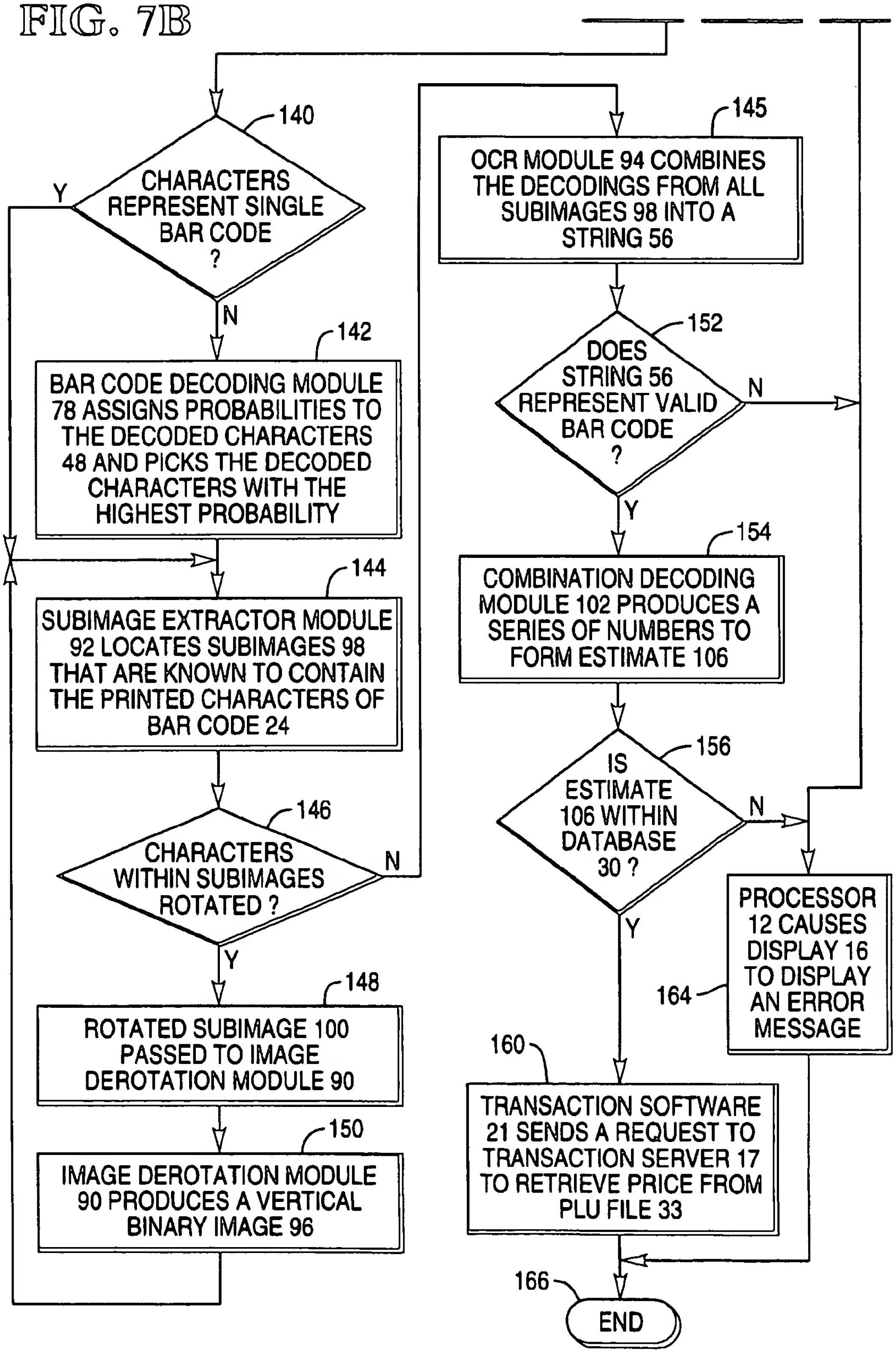

110
START
112
CAMERA 14 PRODUCES 24-BIT COLOR IMAGE 28 OF BAR CODE 24 AND ITEM 26
114
COLOR TO GRAY SCALE CONVERSION MODULE 60 TRANSFORMS IMAGE 28 INTO AN 8-BIT PER PIXEL IMAGE
116
EDGE DETECTION MODULE 62 PRODUCES GRAY SCALE EDGE MAP 68
120
IS DENSE CONCENTRATION OF HIGH-STRENGTH EDGES PRESENT ?
N
Y
124
DO EDGES HAVE A SIMILAR PHASE ?
N
Y
126
PHASE ANALYSIS MODULE 66 DETERMINES ORIENTATION 44 OF BAR CODE 24
128
BAR CODE TRAVERSING MODULE 70 PRODUCES INDIVIDUAL GRAY SCALE SCAN LINES 80
130
THRESHOLD SELECTION ALGORITHM 72 PROVIDES THRESHOLD 86
132
IMAGE THRESHOLDING MODULE 74 PRODUCES THRESHOLDED SCAN LINES 82
134
SCAN LINE CORRELATION AND ENUMERATION MODULE 76 PRODUCES BINARY IMAGE 54 AND A COMPLETE SET OF BINARY LINES 84
136
SCAN LINES 80 DECODEABLE ?
N
Y
138
BAR CODE DECODING MODULE 78 PRODUCES BAR CODE CHARACTERS, BAR CODE TYPE 50, BAR CODE DIRECTION 52

FIG. 7B



140
CHARACTERS REPRESENT SINGLE BAR CODE ?
Y
N
142
BAR CODE DECODING MODULE 78 ASSIGNS PROBABILITIES TO THE DECODED CHARACTERS 48 AND PICKS THE DECODED CHARACTERS WITH THE HIGHEST PROBABILITY
144
SUBIMAGE EXTRACTOR MODULE 92 LOCATES SUBIMAGES 98 THAT ARE KNOWN TO CONTAIN THE PRINTED CHARACTERS OF BAR CODE 24
146
CHARACTERS WITHIN SUBIMAGES ROTATED ?
N
Y
148
ROTATED SUBIMAGE 100 PASSED TO IMAGE DEROTATION MODULE 90
150
IMAGE DEROTATION MODULE 90 PRODUCES A VERTICAL BINARY IMAGE 96
145
OCR MODULE 94 COMBINES THE DECODINGS FROM ALL SUBIMAGES 98 INTO A STRING 56
152
DOES STRING 56 REPRESENT VALID BAR CODE ?
N
Y
154
COMBINATION DECODING MODULE 102 PRODUCES A SERIES OF NUMBERS TO FORM ESTIMATE 106
156
IS ESTIMATE 106 WITHIN DATABASE 30 ?
N
Y
164
PROCESSOR 12 CAUSES DISPLAY 16 TO DISPLAY AN ERROR MESSAGE
160
TRANSACTION SOFTWARE 21 SENDS A REQUEST TO TRANSACTION SERVER 17 TO RETRIEVE PRICE FROM PLU FILE 33
166
END

VISUAL BAR CODE RECOGNITION METHOD

BACKGROUND OF THE INVENTION

The present invention relates to image processing methods and bar code scanners, and more specifically to a visual bar code recognition method.

Bar code symbols provide a fast and accurate means of representing information about an object. Decoding or reading of the bar code is accomplished by translating the patterns of bars and spaces into a unique series of numbers that correspond to a specific item. Currently, the majority of bar codes are read using laser scanners. While these laser systems work well under optimal conditions, they have some inherent disadvantages and limitations. For example, a laser scanner cannot verify that a correct bar code has been scanned by an examination of the item.

Therefore, it would be desirable to provide a visual bar code recognition method, which may be part of a visual bar code scanner.

SUMMARY OF THE INVENTION

In accordance with the teachings of the present invention, a visual bar code recognition method is provided.

The visual bar code recognition method captures a color image of an object containing a bar code. Regardless of the orientation of the bar code within the field-of-view, the system detects the presence of the bar code, and decodes it using the bar/space patterns. It then produces an independent decoding of the human-readable numbers printed on the bar code using OCR. From these two decodings, it determines the identity of the object. It verifies this identity by comparing the physical characteristics of the object from the image with the known features of the product.

The method adds a great deal of decoding flexibility not possible with a laser since a laser scanner can only perform a subset of these tasks.

As long as the bar code is within the focused field of view, the algorithms are capable of locating and decoding the bar code regardless of orientation.

Error detection and correction are provided in two ways: (a) Multiple scan lines are passed through the bar code allowing for partial decoding of several lines and recombining into a final result and (b) Optical Character Recognition (OCR) on the characters below the bar code normally used for manual entry allow for an independent decoding.

After decoding, products may be verified by comparing the known physical characteristics (color, size, shape, texture, etc.) of the decoded product with the features found in the captured image.

The method is applicable to both processing on a single frame static image (such as with a hand-held camera) or on a real-time image stream. The implementations of these ideas are different in each case but the underlying process is identical.

It is accordingly an object of the present invention to provide a visual bar code recognition method.

It is another object of the present invention to provide a visual bar code recognition method which uses a camera to capture an image of an item having a bar code.

It is another object of the present invention to provide a visual bar code recognition method which uses a camera to capture an image of an item having a bar code, and which locates and decodes the bar code.

It is another object of the present invention to provide a visual bar code recognition method which uses a camera to capture an image of an item having a bar code, and which verifies the contents of the bar code using optical character recognition (OCR).

It is another object of the present invention to provide a visual bar code recognition method which uses a camera to capture an image of an item having a bar code, and which verifies the contents of the bar code by comparing features of the item from the image with features in a product database.

It is another object of the present invention to provide a visual bar code recognition method which uses a camera to capture an image of an item having a bar code which may be static or moving.

BRIEF DESCRIPTION OF THE DRAWING

Additional benefits and advantages of the present invention will become apparent to those skilled in the art to which this invention relates from the subsequent description of the preferred embodiments and the appended claims, taken in conjunction with the accompanying drawings, in which:

FIG. **1** is a block diagram of a visual bar code system of the present invention;

FIG. **2** is a block diagram of the bar code reading program of FIG. **1**;

FIG. **3** is a block diagram of the bar code location and orientation module of FIG. **2**;

FIG. **4** is a block diagram of the scan line extraction and decoding module of FIG. **2**;

FIG. **5** is a block diagram of the optical character recognition (OCR) decoding module of FIG. **2**;

FIG. **6** is a block diagram of the combination and verification module of FIG. **2**; and

FIGS. **7A** and **7B** form a flow diagram of the visual bar code decoding method.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring now to FIG. **1**, system **10** primarily includes camera **14**, terminal **16**, and transaction server **17**.

Camera **14** produces an image **28** of bar code **24** and item **26**. Preferably, image **28** is a 24-bit color image. Bar code **24** includes black and white bars and human-readable characters. Item **26** may be static or moving.

Terminal **16** includes processor **12**.

Processor **12** executes transaction processing software **21** which tallies items during a transaction, including item **26**. In order to obtain price information for an item, transaction processing software **21** sends a price request containing an item number obtained from bar code reading software **22** to transaction server **17**.

Bar code reading software **22** locates and decodes bar code **24** and sends the item number to transaction processing software **21**. Bar code reading software **22** produces decoded bar code information by analyzing image **28**, by using optical character recognition of numeric characters printed with the bar code and evident in image **28**, and by comparing features extracted from image **28** with features stored within product database **30**.

Transaction server **17** provides item price and item descriptions from PLU file **33** in response to requests from terminal **16**.

Storage medium **18** permanently stores bar code reading software **22** and contains product database **30**. Product

database 30 contains item features that bar code reading software 22 uses to identify item 26. Thus, only items whose features have been previously entered in product database 30 are identifiable.

Display 20 displays item price and item descriptions 32 provided by processor 12.

Storage medium 31 stores PLU file 33.

Turning now to FIG. 2, bar code reading software 22 includes bar code location and orientation determining module 34, scan line extraction and decoding module 36, optical character recognition (OCR) decoding module 38, and combination and verification module 40.

Bar code location and orientation determining module 34 determines the location 42 and orientation 44 of bar code 24 and produces a gray scale image 46 from color image 28. Bar code location and orientation determining module 34 utilizes the highly parallel nature of the bars within bar code 24. In an image such as image 28, the edges or boundaries of these parallel bars will themselves be parallel, pointing in a direction perpendicular to the bars. The presence of a compact set of unidirectional edges signals the possibility of a bar code. In frequency space, the predominance of a single phase of these edges filters a bar code from surrounding text and determines the orientation of the bar code. This edge-based approach to bar code location and orientation is valid in both a static image (such as one produced using a hand-held camera) or a frame from an real-time image stream.

Scan line extraction and decoding module 36 uses location 42, orientation 44, and gray scale image 46 to decode bar code 24 from its bar and space patterns and produces decoded bar code characters string 48. By-products of scan line decoding are bar code type 50, bar code direction 52, and binary image 54.

OCR decoding module 38 uses bar code type 50, bar code direction 52, and binary image 54, along with bar code location 42 and orientation 44, to extract the precise regions in image 46 that contain the human-readable characters of bar code 28 and produce decoded human-readable characters string 56.

Combination and verification module 40 produces a final decoding 58 of bar code 24 from three sources: decoded bar code characters string 48, decoded human-readable character string 56, and features in image 28.

Turning now to FIG. 3, bar code location and orientation determining module 34 includes color to gray scale conversion module 60, edge detection module 62, magnitude analysis module 64, and phase analysis module 66.

Color to gray scale conversion module 60 transforms image 28 into an 8-bit per pixel representation, which is gray scale image 46.

Edge detection module 62 applies a filter at each pixel of gray scale image 46 to produce a gray scale edge map 68 that indicates whether each point of image 46 is a member of the set of bar code edge pixels (the stronger the edge in the image, the greater the gray level in edge map 68.) The filter filters out pixels having gray scale levels below a predetermined threshold gray scale level.

Magnitude analysis module 64 analyzes edge map 68 to provide the location of bar code 24. An area of edge map 68 with a dense concentration of high strength edges indicates a region of good contrast and the likely location of bar code 24. Magnitude analysis module 64 looks for a density of pixels left from the filtering by edge detection module 62 that is greater than a predetermined threshold density. The center of the region is termed the location 42 of bar code 24.

Phase analysis module 66 employs location information 42 from magnitude analysis module 64 and edge map 68 to determine the orientation 44 of bar code 24. If the phase or direction is similar for most of the high strength edges, the region probably contains the parallel bars of bar code 24. The phase of the edges is the bar code's orientation 44.

Referring now to FIG. 4, scan line extraction and decoding module 36 includes bar code traversing module 70, threshold selection algorithm 72, image thresholding module 74, scan line correlation and enumeration module 76, and bar code decoding module 78.

Bar code traversing module 70 traces a series of gray scale scan lines 80 completely across the bar code 24 at the computed orientation angle 44. This bar code traversal is comprised of three steps: defining coordinates that make up a line at the given angle (with origin at (0,0)), determining starting points at both sides of the bar code (since its direction is unknown at this stage), and creating a set of scan lines 80 using each starting point (from Step 2) as an offset to the line determined in Step 1.

Due to the design of the bar code itself, it is not necessary for this angle 44 to be exact. All that is required is that each scan line 80 passes completely through the bar code 24. The coordinates in Step 1 may be computed as needed or predetermined and stored in a lookup table.

Threshold selection algorithm 72 transforms the gray scale image 46 into a binary image 54. This is accomplished by selecting a single value or threshold and mapping all pixels whose gray levels are greater than the threshold to one and all those below the threshold to zero.

The literature supports a large number of threshold selection algorithms 72 and the results of this bar code decoding method are not dependent on any particular method. Experimentally, numerical techniques appear to be preferable to statistical (histogram) methods. To allow for a margin of error when thresholding, it may be desirable to select more than one threshold (usually by varying parameters in a single thresholding scheme). The threshold used to create the binary image 54 is either the single computed value or the average of several.

Imaging thresholding module 74 transforms gray scale scan lines 80. If N thresholds are selected, the thresholded output contains N+1 levels. Therefore, the output of imaging thresholding module 74 may be of binary, ternary, or higher order.

Scan line correlation and enumeration module 76 produces a complete set of binary scan lines 84. The correlation part of this step combines groups of M adjacent lines in a effort to reduce the effects of noise and thresholding artifacts. From these combined scan lines, the enumerator portion constructs a set of binary scan lines 84 representing all possible pixel patterns.

Bar code decoding module 78 decodes each of scan lines 84 by measuring the bar/space patterns and translating them into a string 48 containing the bar code characters along with the bar code type 50 (UPC-A, UPC-E, etc.) and the bar code direction 52 (left to right or right to left).

Bar code decoding module 78 additionally verifies that each decoded string 48 satisfies the checksum requirements for a bar code. If errors are found in one portion of a scan line, bar code decoding module 78 will attempt to salvage any information possible from sections of the line. Ideally, all of the scan lines will decode to the same bar code. If not, bar code decoding module 78 then assigns probabilities to the different decodings based on the number of scan lines producing each. The algorithms developed to decode laser scanned bar codes are applicable here.

With reference to FIG. 5, OCR decoding module 38 includes image derotation module 90, subimage extraction module 92, and OCR module 94.

Subimage extractor 92 appropriately locates the subimages 98 of binary image 54 that contain the printed human-readable characters. For instance, UPC-A has ten characters, separated into two groups of five, printed directly below the bars (inside the guard bars), one on the bottom left and one on the bottom right. Subimage extractor 92 places rectangles at these four locations and isolates the pixels within these boxes.

OCR module 94 then reads the characters from subimages 98, choosing only numbers as possible characters. It then combines the decodings from all subimages 98 into a single bar code string 56, again validating the checksum.

Image derotation module 90 produces a binary image 96 in which bar code 24 is guaranteed to be vertical. This is necessary if OCR module 94 cannot handle rotated text in subimages 98. Rotated subimages 100 are passed through OCR module 94 to image derotation module 90, as discussed further below.

If a single character decodes as more than one number, both selections are retained and probabilities are assigned to each. Decoding of such numbers is resolved by combination and verification module 40.

With reference to FIG. 6, combination and verification module 40 includes combination decoding module 102 and product verification module 104.

Combination decoding module 102 compares strings 48 and 56 to determine an estimate 106 of the identity of item 26.

Product verification module 104 compares features identified within image 28 with features stored within product database 30 to estimate the identity of item 26. Features include, but are not limited to, such attributes as shape, size, and color scheme of item packaging, and text and logos printed on the item or the item packaging.

Since both the bar code string 48 and the OCR string 56 may have a degree of uncertainty associated with each character, product verification module 104 then compares estimate 106 suggested by strings 48 and 56 with the estimate determined by product verification module 104. Uncertainty may be caused by one or more characters being undecodable using OCR decoding module 38 or by the bar/space patterns being undecodable. Therefore, it is entirely possible that neither method of modules 36 and 38 can decode bar code 24 correctly. Product verification module 104 produces the final decoding 58 of item 26 based on these probabilities. The final decoding 56 is a series of numbers that corresponds to item 26 in the product database 30.

Turning now to FIGS. 7A and 7B, the method of operation of system 10 and software 22 is illustrated beginning with START 110.

In step 112, camera 14 produces 24-bit color image 28 of bar code 24 and item 26. Item 26 may be static or moving in front of camera 14.

In step 114, color to gray scale conversion module 60 transforms image 28 into an 8-bit per pixel gray scale image 46. This step decreases memory and processing power that are required to locate bar code 24.

In step 116, edge detection module 62 produces gray scale edge map 68.

In step 120, magnitude analysis module 64 attempts to locate bar code 24. If an area characterized by a dense concentration of high-strength edges is not found, then item 26 is not properly oriented or printed or does not have a bar code. The method proceeds to step 164, in which software 22 displays an error message on display 20 and the method ends in step 166.

If image 46 contains a dense concentration of high-strength edges, the method proceeds to step 124.

In step 124, phase analysis module 66 determines whether the edges have a similar phase so that the area found in step 120 may be classified as a bar code.

If not, then the area is not a bar code or bar code 24 is so poorly printed that decoding is not possible. The method proceeds to step 164, in which software 22 displays an error message on display 20 and the method ends in step 166.

If the edges have a similar phase, then the area is likely a bar code and the method proceeds to step 126.

In step 126, phase analysis module 66 determines orientation 44 of bar code 24.

In step 128, bar code traversing module 70 produces individual gray scale scan lines 80.

In step 130, threshold selection algorithm 72 provides threshold 86.

In step 132, image thresholding module 74 produces thresholded scan lines 82.

In step 134, scan line correlation and enumeration module 76 produces binary image 54 and a complete set of binary lines 84.

In step 136, bar code decoding module 78 attempts to decode binary lines 84.

If there are errors, the method proceeds to step 164, in which software 22 displays an error message on display 20 and the method ends in step 166.

If there are no errors, then the method proceeds to step 138.

In step 138, bar code decoding module 78 produces bar code characters, bar code type 50, and bar code direction 52.

In step 140, bar code decoding module 78 determines whether the characters represent a single bar code.

If so, the method proceeds to step 144.

If not, the method proceeds to step 142.

In step 142, bar code decoding module 78 assigns probabilities to the decoded characters and picks the decoded characters with the highest probabilities to form string 48.

In step 144, subimage extractor module 92 locates subimages 98 that are known to contain the printed characters of bar code 24.

In step 146, subimage extractor module 92 determines whether characters within the subimages are rotated.

If not, the method proceeds to step 145.

If so, the method proceeds to step 148.

In step 148, subimage extractor module 92 passes the rotated subimage 100 to image derotation module 90.

In step 150, image derotation module 90 produces a vertical binary image 96 and the method returns to step 144 until all of the subimages are derotated.

In step 145, OCR module 94 combines the decodings from all subimages 98 through OCR module 94 into string 56.

In step 152, OCR module 94 determines whether string 56 represents a valid bar code.

If not, the method proceeds to step 164, in which software 22 displays an error message on display 20 and the method ends in step 166.

If so, then the method proceeds to step **154**.

In step **154**, combination decoding module **102** compares string **48** with string **56** to produce a series of numbers to form estimate **106**.

In step **156**, product verification module **104** compares features identified within image **28** with features stored within product database **30** to determine whether estimate **106** is in product database **30**.

If not, the method proceeds to step **164**, in which software **22** displays an error message on display **20** and the method ends in step **166**.

If so, then the method proceeds to step **160**.

In step **160**, transaction software **21** sends a request to transaction server **17** to retrieve the price of item **26** from PLU file **33**. Transaction software **21** adds the item and its price to the transaction total and completes the transaction after all such items have been processed by software **22**.

The method ends in step **166**.

Although the present invention has been described with particular reference to certain preferred embodiments thereof, variations and modifications of the present invention can be effected within the spirit and scope of the following claims.

What in claimed is:

1. A method of decoding a bar code to identify an item having the bar code on it comprising the steps of:

capturing an image of the item;

locating the bar code label in an area of the image;

decoding the bar code label located in the area of the image to produce a first set of characters;

performing optical character recognition of the area of the image to produce a second set of characters; and

comparing the first set of characters to the second set of characters to identify the item.

2. The method as recited in claim **1**, further comprising the steps of:

determining predetermined features of the item from the image; and

comparing the features to features stored within a database to verify the identity of the item.

3. A method of decoding a bar code on an item comprising the steps of:

providing a gray-scale image of the item;

determining gray levels of pixels within the image;

filtering out pixels having gray levels below a predetermined threshold gray level;

determining an area of the image having a density of pixels with gray levels above the predetermined threshold density to locate the bar code;

determining an orientation of the area;

tracing a plurality of gray-scale lines through the area;

transforming the image into a binary image;

transforming the gray-scale lines to binary lines;

decoding the binary lines to produce a first set of characters;

performing optical character recognition of the area to produce a second set of characters;

comparing the first set of characters to the second set of characters to identify the item;

determining predetermined features of the item from the image; and comparing the features to features stored within a database to verify the identity of the item.

4. The method of claim **1** wherein the step of capturing an image of the item is performed utilizing a camera to scan a substantial portion of the item including the bar code.

5. The method of claim **1** wherein the step of locating the bar code label further comprises the steps of determining the location of a dense concentration of high-strength edges within the image and analyzing whether said edges have a similar phase.

6. The method of claim **1** wherein the bar code has an associated human readable character string located in close proximity to a series of bars and spaces defining a coded portion of the bar code, said step of decoding the bar code label comprises a decoding of the series of bars and spaces, and said step of performing optical character recognition comprises a recognition of said associated human readable character string.

7. The method of claim **6** wherein identification of the item occurs only if the first set of characters and the second set of characters sufficiently match.

8. The method of claim **2** wherein said predetermined features correspond to physical characteristics of the item.

9. The method of claim **8** wherein identification of the item occurs only if the first set of characters and the second set of characters sufficiently match and said predetermined features sufficiently match the features stored within the database.

10. A visual recognition apparatus for decoding a bar code to identify an item, the apparatus comprising:

a camera for capturing an image of the item;

a bar code label locator for locating an area within the image including the bar code;

a decoder for decoding the bar code and to produce a first set of characters corresponding to the bar code;

an optical character recognizer for recognizing a human-readable string of characters associated with the bar code and also located in said area, and for producing a second set of characters corresponding to the human-readable string of characters; and

a processor for comparing the first set of characters and the second set of characters and to identify the item if the first set of characters and the second set of characters sufficiently match.

11. The apparatus of claim **10** wherein the processor is further operable to determine predetermined features of the item from the image, and to compare the predetermined features to features stored within a database to further verify the identity of the item.

12. The apparatus of claim **10** wherein the bar code label locator further comprises an edge filter for locating an area within the image having a dense concentration of high-strength edges.

13. The apparatus of claim **12** further comprising an analyzer for analyzing whether said edges have a similar phase.

14. The apparatus of claim **11** wherein the predetermined features correspond to physical features of the item.

* * * * *

103 Claim 14

US005969756A

United States Patent [19]

Buckley et al.

[11] **Patent Number: 5,969,756**

[45] **Date of Patent: *Oct. 19, 1999**

[54] **TEST AND ALIGNMENT SYSTEM FOR ELECTRONIC DISPLAY DEVICES AND TEST FIXTURE FOR SAME**

[75] Inventors: **Eric Buckley**, Scarborough; **Branko Bukal**, Thornhill; **Wayne Dawe**, Richmond Hill; **Paul Farrer**, Scarborough; **Karoly G. Nemeth**, Don Mills; **Andrew Noonan**, Oshawa, all of Canada

[73] Assignee: **Image Processing Systems Inc.**, Ontario, Canada

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **08/670,694**

[22] Filed: **Jun. 26, 1996**

Related U.S. Application Data

[63] Continuation-in-part of application No. PCT/CA95/00352, Jun. 13, 1995, which is a continuation-in-part of application No. 08/259,309, Jun. 13, 1994, abandoned.

[51] **Int. Cl.**[6] **H04N 17/00**; H04N 17/02; H04N 17/04

[52] **U.S. Cl.** **348/190**; 348/191

[58] **Field of Search** 348/180, 181, 348/184, 187, 188, 189, 190, 191; H04N 17/04, 17/00, 17/02

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,962,722	6/1976	Ciciora	358/10
4,330,779	5/1982	Wilensky et al.	340/705
4,415,921	11/1983	Mulvanny et al.	358/139
4,575,753	3/1986	Mistry et al.	358/118
4,688,079	8/1987	Fendley	358/10
4,751,570	6/1988	Robinson	358/88
4,754,329	6/1988	Lindsay et al.	358/139
4,760,444	7/1988	Nielson et al.	358/107
4,814,858	3/1989	Mochizuki et al.	348/191
4,893,925	1/1990	Sweeny et al.	356/72
5,032,769	7/1991	Kawakami	315/368
5,049,791	9/1991	Kawakami	315/368
5,216,504	6/1993	Webb et al.	358/139
5,371,537	12/1994	Bohan et al.	348/181

FOREIGN PATENT DOCUMENTS

404700	12/1990	European Pat. Off.	H04N 17/04
0 577 380 A1	1/1994	European Pat. Off. .	
WO95/34906	12/1995	WIPO	H04N 17/04

OTHER PUBLICATIONS

Kim, S.R. et al; "Design sand Implementation of an Automatic Adjustment System for Integrated Tube Components" *Mechatronics*, vol. 4, No. 1, pp. 1–23; Oxford, Great Britian.

Lin, Robert Jr.; "Automated CRT inspection and Alignment"; *Information Display*; vol. 4, No. 6, 8, pp. 16–17.

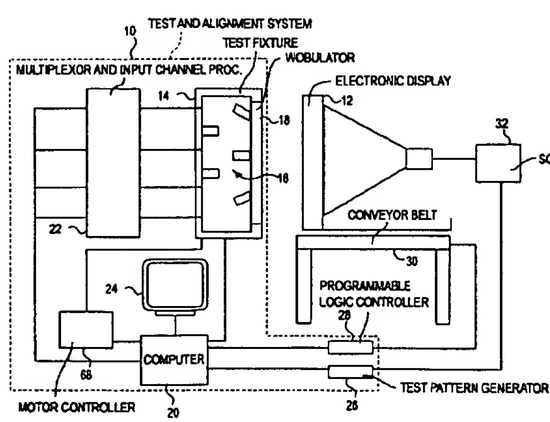
Hibara, T. et al, "Automatic adjustment for color display monitor"; Proceedings of 1986 International Conference on Industrial Electronics, Control & Instrumentation (IECON), Milwaukee, WI, pp. 164–169.

Primary Examiner—Michael Lee
Attorney, Agent, or Firm—Venable; Robert Kinberg

[57] **ABSTRACT**

A test and alignment system for an electronic display device comprises a test pattern generator to be connected to an electronic display device for causing images of video test patterns to be displayed by the electronic display device. A test fixture is positioned in front of the electronic display device to be tested and aligned. The test fixture includes a frame supporting a plurality of close-up optical sensors to sense and produce image signals corresponding to small areas of images displayed on the electronic display device and a plurality of wide-angle optical sensors behind the close-up optical sensors for sensing and producing image signals corresponding to large areas of images displayed on the electronic display device. A computer controls the test pattern generator and processes and analyses the image signals generated by the close-up and wide-angle optical sensors to perform a series of tests on the electronic display device. A display provides a visual indication of the results of the series of tests performed by the computer.

37 Claims, 10 Drawing Sheets



12/17/2003, EAST Version: 1.4.1

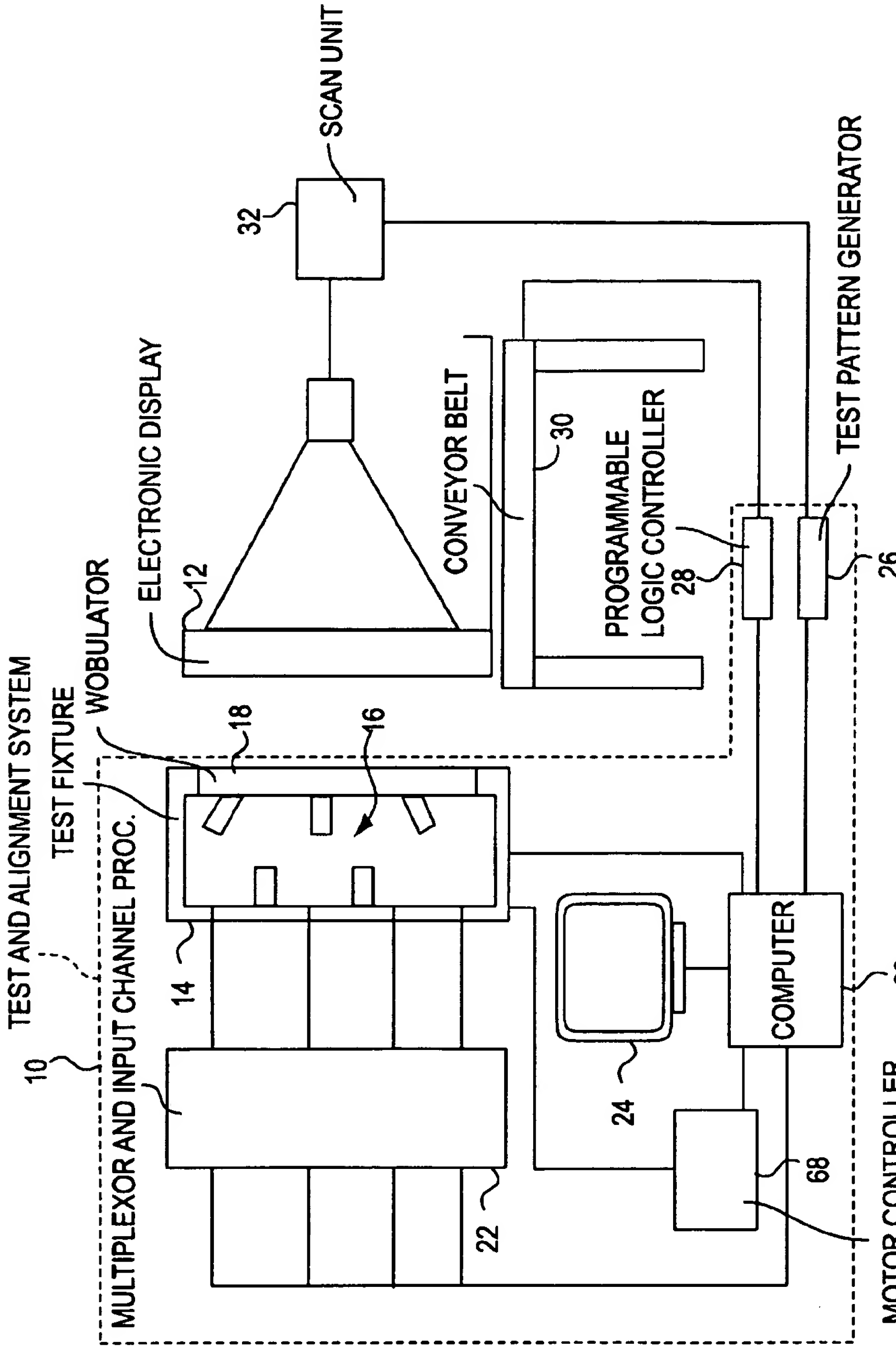

SCAN UNIT
32
ELECTRONIC DISPLAY
12
CONVEYOR BELT
30
PROGRAMMABLE
LOGIC CONTROLLER
28
TEST PATTERN GENERATOR
26
TEST AND ALIGNMENT SYSTEM
TEST FIXTURE
WOBULATOR
18
16
10
14
MULTIPLEXOR AND INPUT CHANNEL PROC.
COMPUTER
20
24
68
22
MOTOR CONTROLLER

FIG. 1

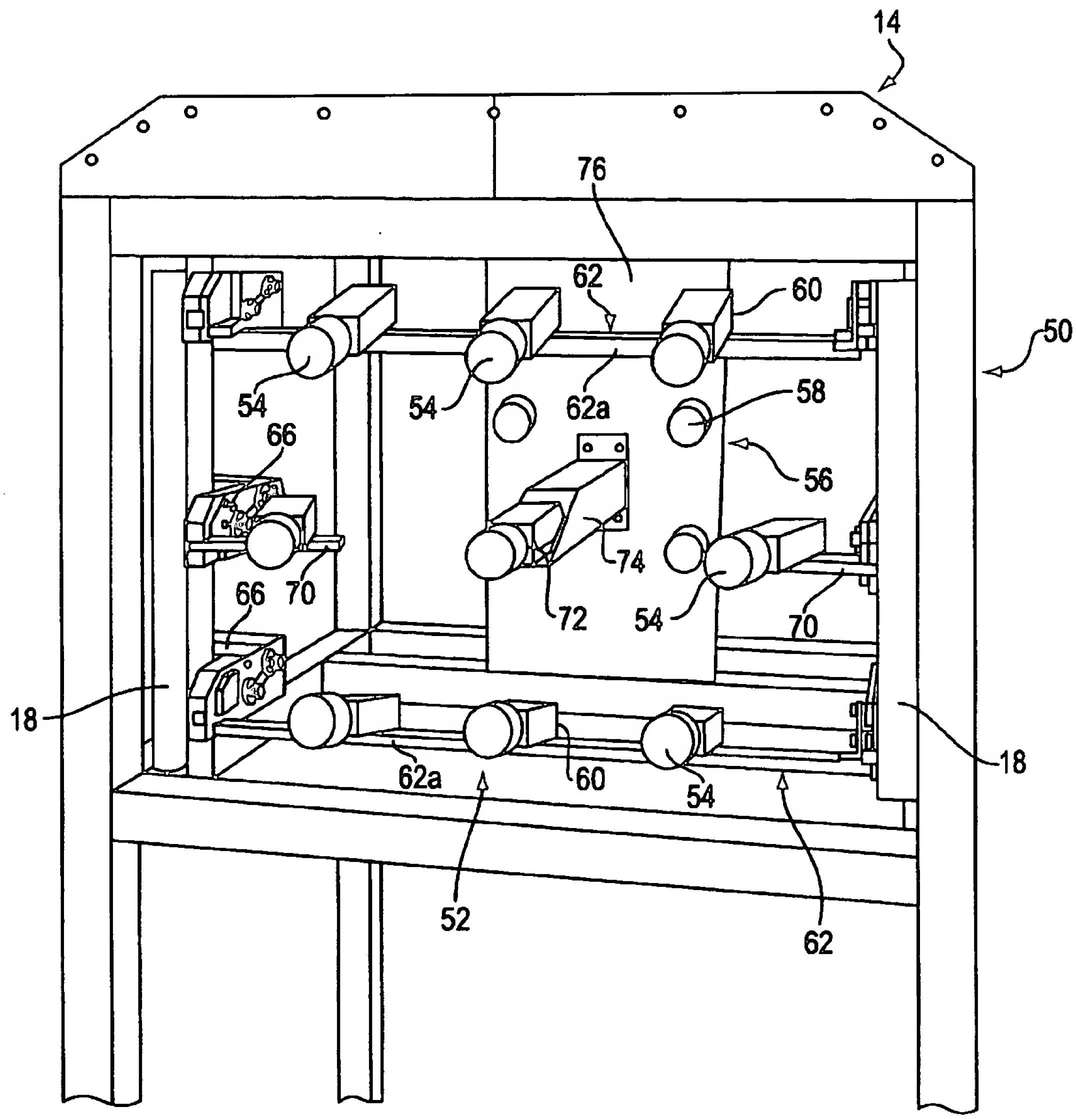

14
76
62
60
50
62a
54
54
58
62a
66
56
74
66
70
72
54
70
18
18
62a
60
54
52
62

FIG. 2a

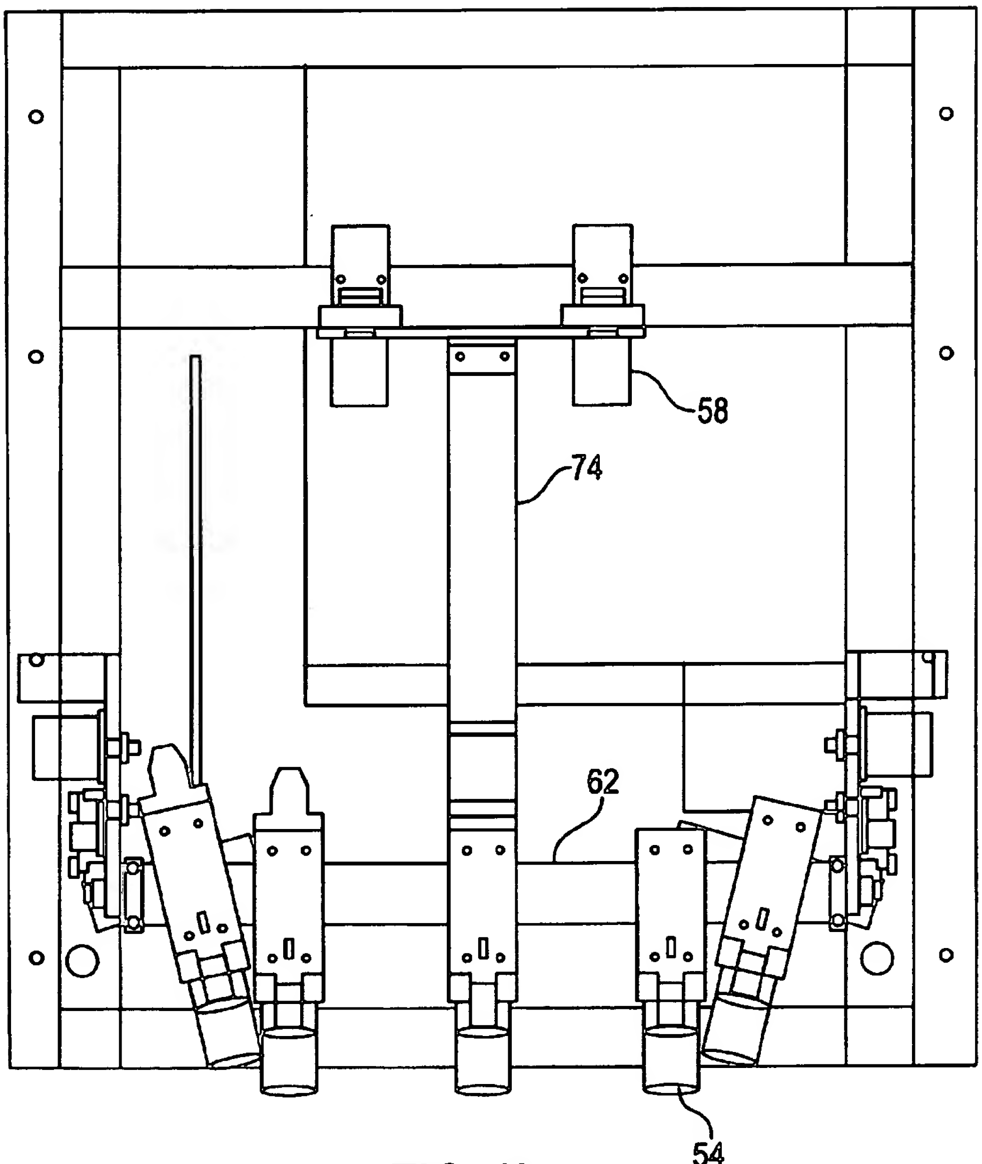


FIG. 2b

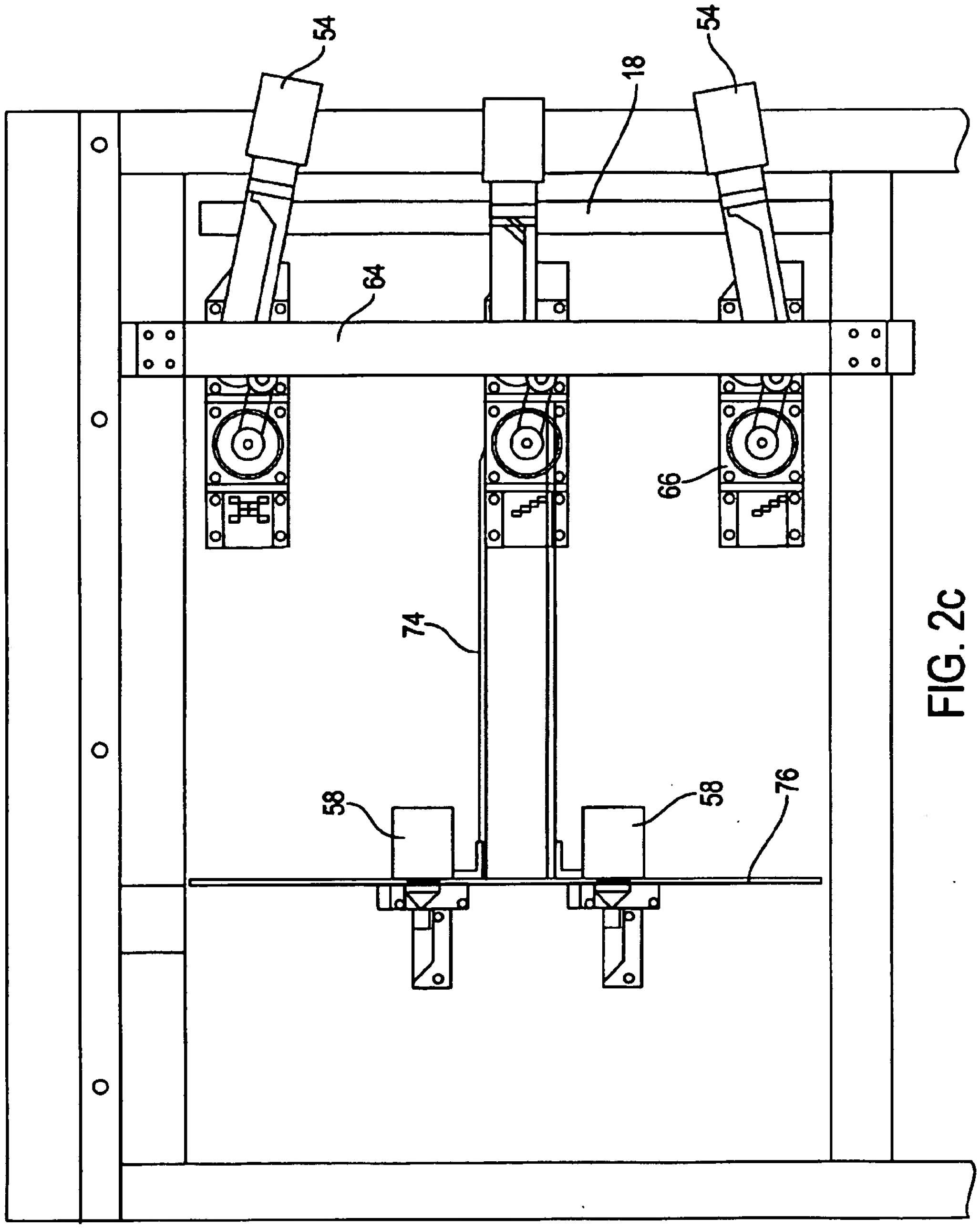


FIG. 2c

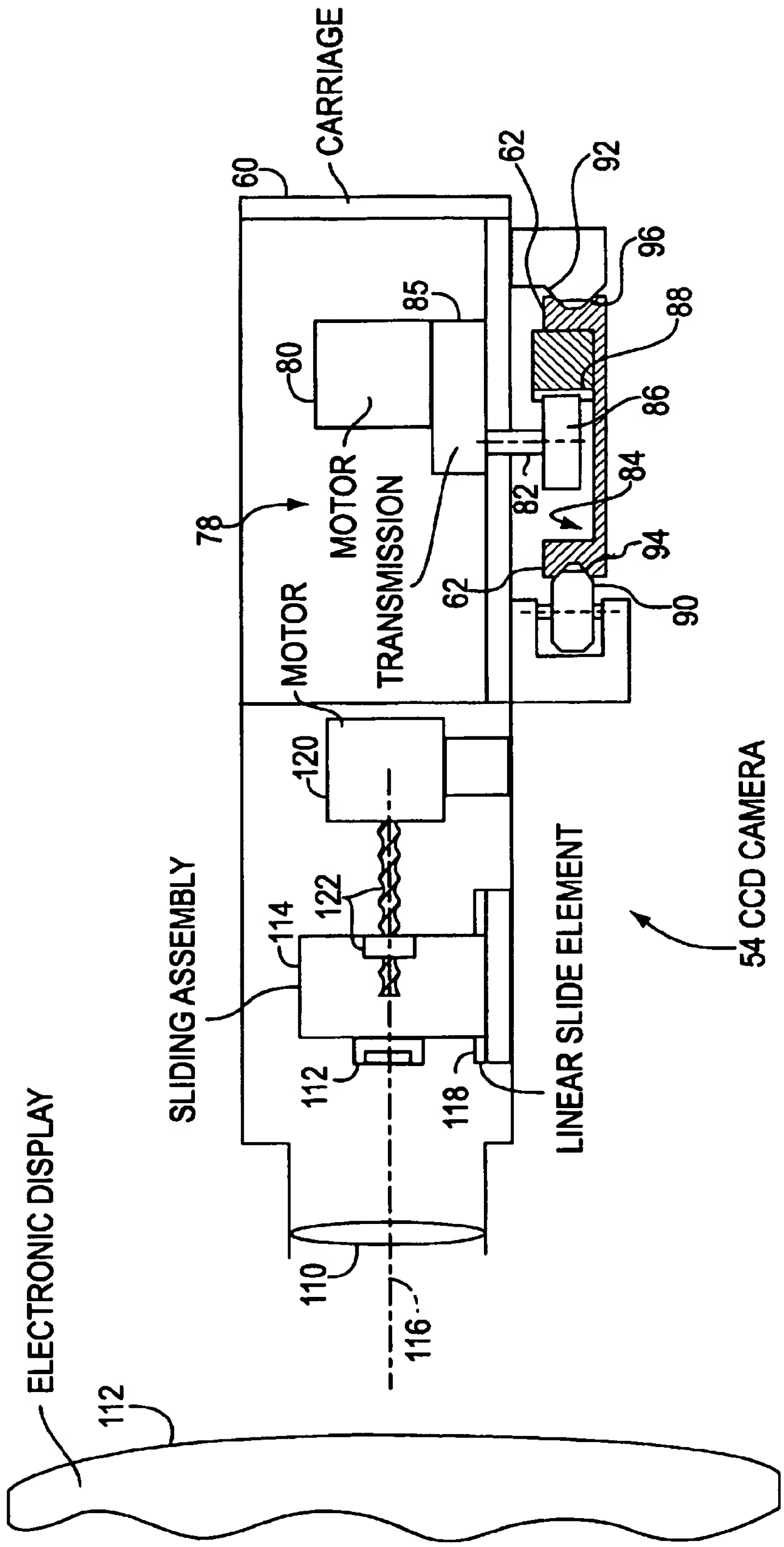
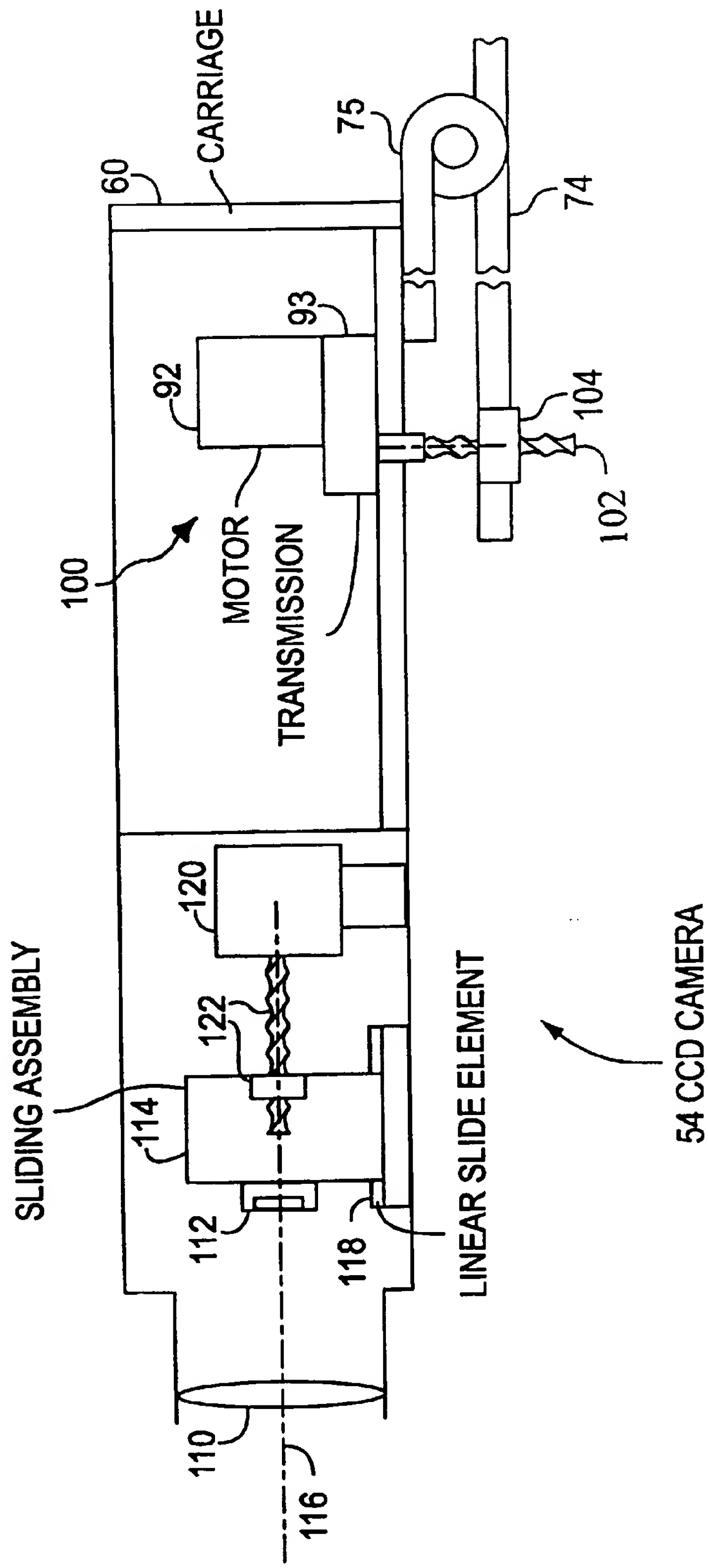

CARRIAGE
60
62
92
96
88
85
80
86
MOTOR
TRANSMISSION
82
84
78
62
94
90
MOTOR
120
54 CCD CAMERA
LINEAR SLIDE ELEMENT
SLIDING ASSEMBLY
122
114
112
118
110
116
ELECTRONIC DISPLAY
112

FIG. 3



CARRIAGE
60
75
74
93
92
104
102
100
MOTOR
TRANSMISSION
120
122
SLIDING ASSEMBLY
114
112
118
LINEAR SLIDE ELEMENT
54 CCD CAMERA
110
116

FIG. 4

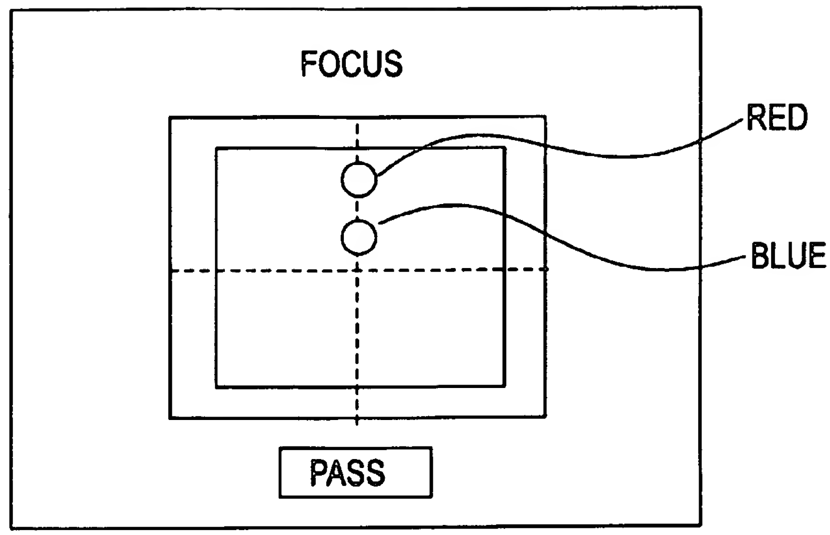

FOCUS
RED
BLUE
PASS

FIG. 5

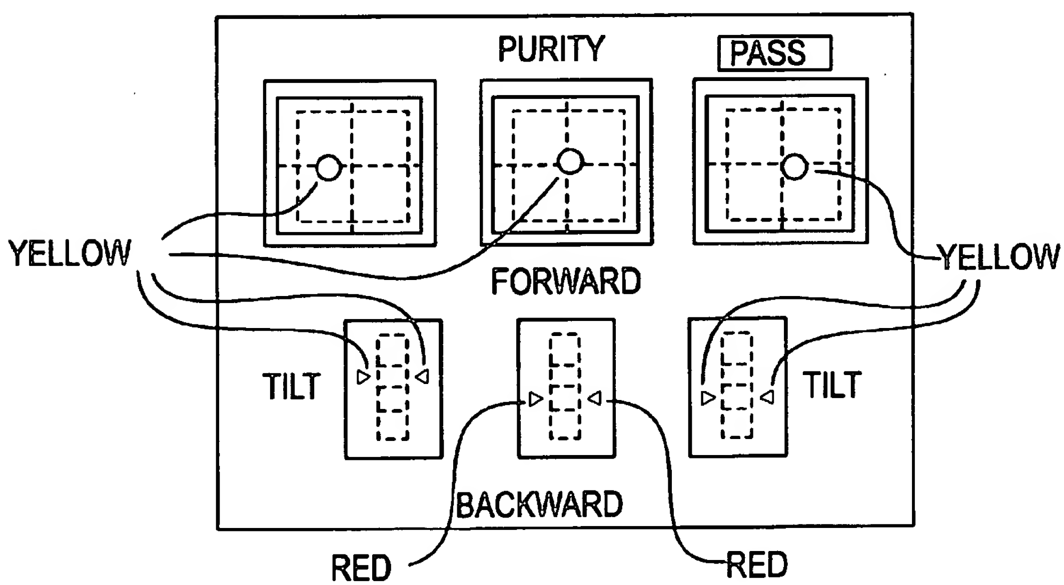

PURITY
PASS
YELLOW
FORWARD
YELLOW
TILT
TILT
BACKWARD
RED
RED

FIG. 6

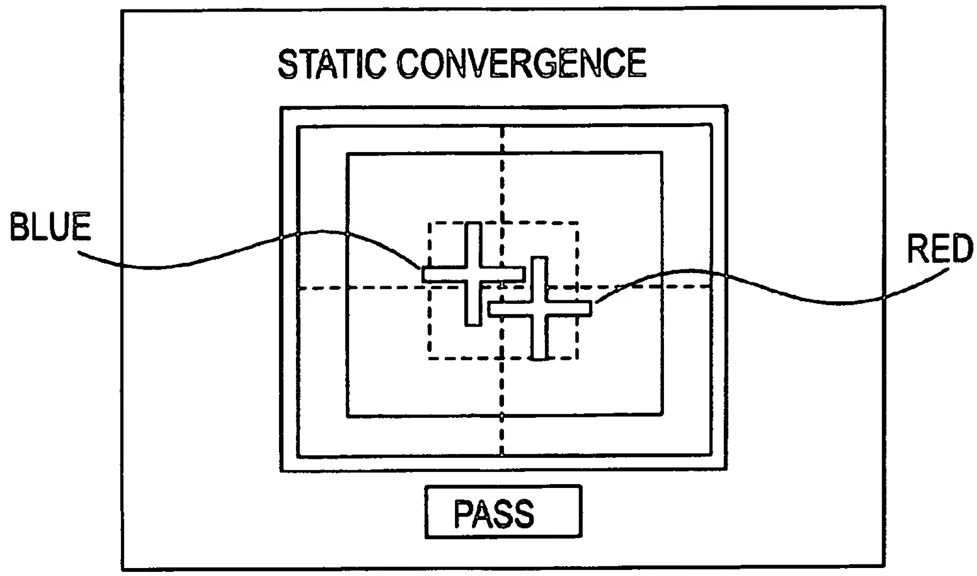

STATIC CONVERGENCE
BLUE
RED
PASS

FIG. 7

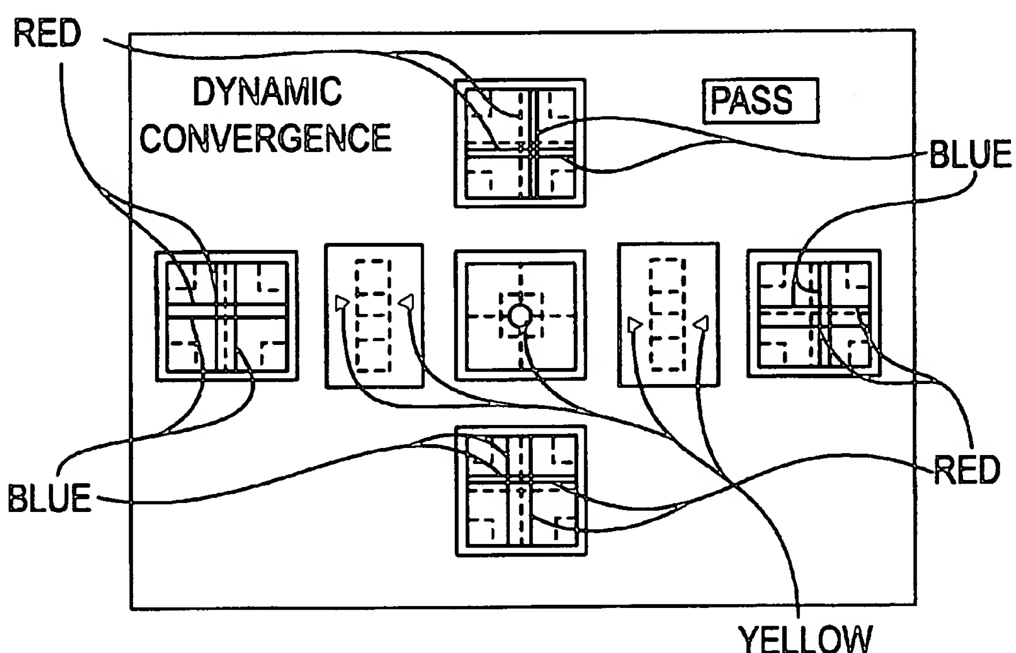

RED
DYNAMIC
CONVERGENCE
PASS
BLUE
BLUE
RED
YELLOW

FIG. 8

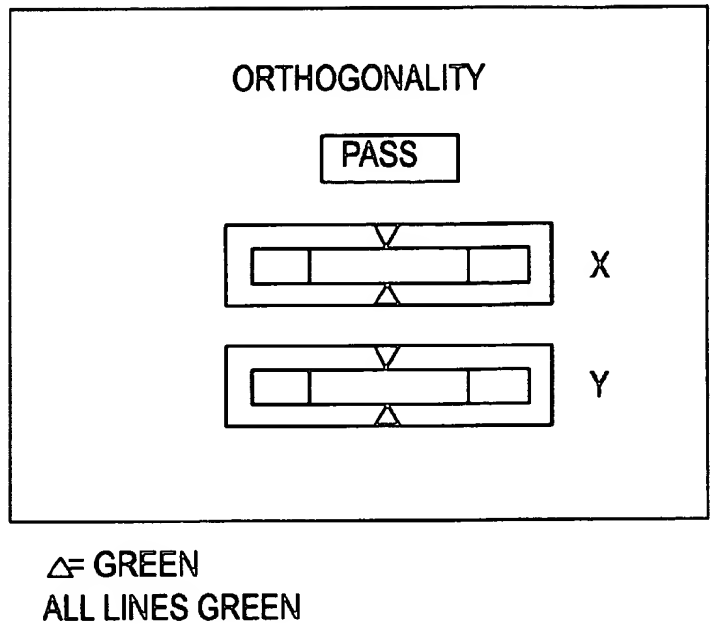

ORTHOGONALITY
PASS
X
Y
△= GREEN
ALL LINES GREEN

FIG. 9

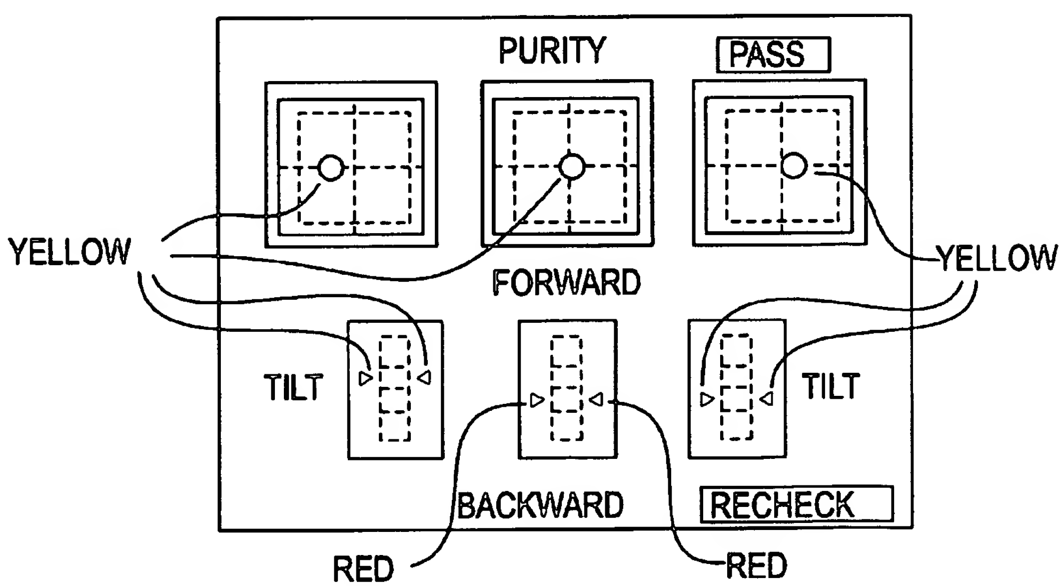

PURITY
PASS
YELLOW
FORWARD
YELLOW
TILT
TILT
BACKWARD
RECHECK
RED
RED

FIG. 10

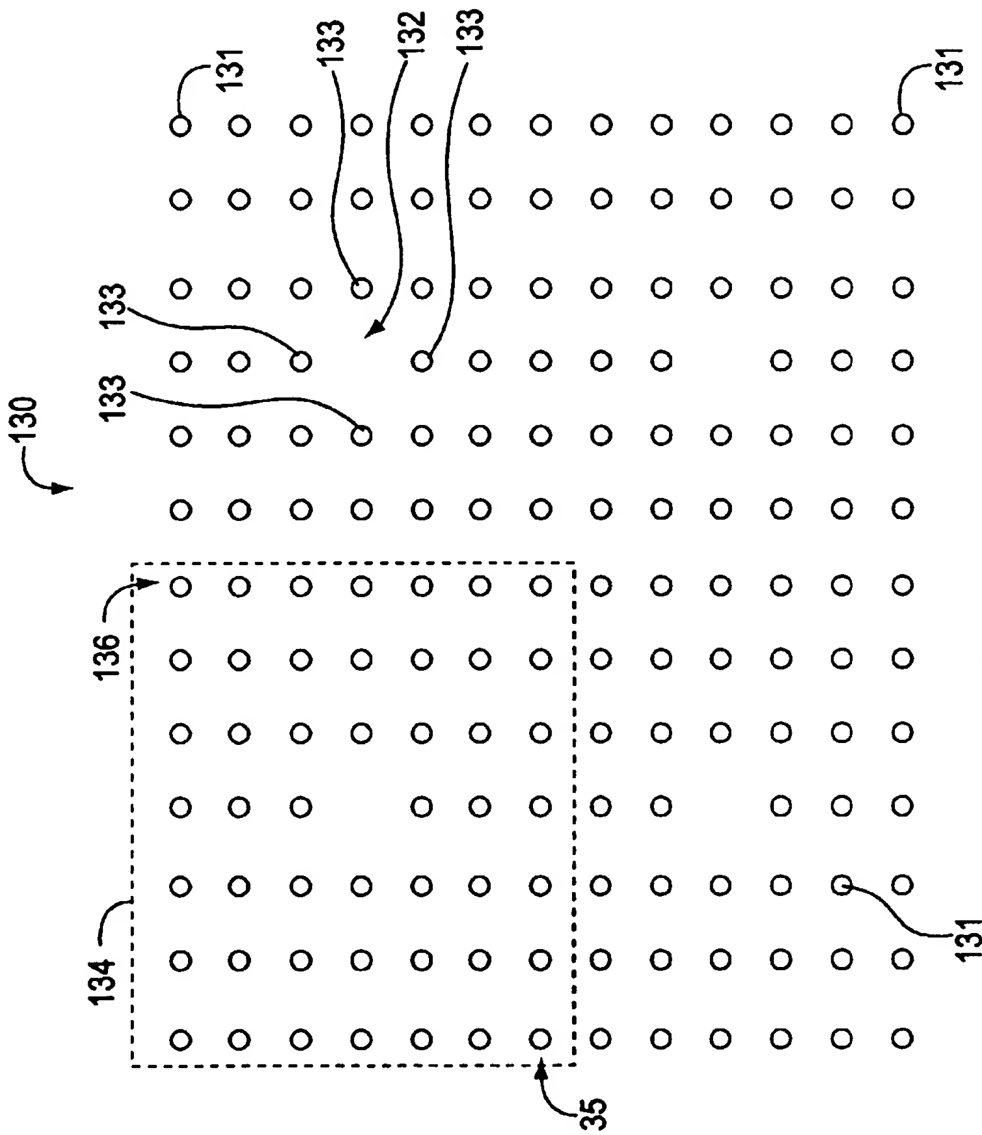


FIG. 11

TEST AND ALIGNMENT SYSTEM FOR ELECTRONIC DISPLAY DEVICES AND TEST FIXTURE FOR SAME

RELATED APPLICATIONS

The present application is a continuation-in-part of PCT application serial No. PCT/CA95/00352 filed on Jun. 13, 1995 designating the United States which is a continuation-in-part of U.S. application Ser. No. 08/259,309 filed on Jun. 13, 1994, now abandoned.

FIELD OF THE INVENTION

The present invention relates to testing systems and in particular to a test and alignment system for an electronic display device and to a test fixture for the same.

BACKGROUND OF THE INVENTION

During the manufacture and assembly of electronic display devices, such as for example cathode ray tube (CRT) assemblies for computer monitors and television sets, precise mechanical, optical and electronic adjustments are required to ensure the electronic display devices provide optimum reproduction image quality. These adjustments include, but are not limited to, focus, purity of color, convergence of beams, color uniformity, geometry, and luminance. Each of these adjustments is typically performed independently of the others by a trained technician with the aid of a testing and alignment system. Unfortunately, many testing and alignment systems are deficient because they are unable to perform all of the tests needed to align precisely a CRT assembly, are very slow and unable to provide real-time feedback for operator adjustments, are not sufficiently accurate and are not integrated to perform each of the tests in an optimized sequence and then recheck and readjust measurements as required.

To deal with the above-described problems, improved test and alignment systems for electronic display devices have been developed. For example, PCT application serial No. PCT/CA95/00352 published on Dec. 21, 1995 and assigned to Image Processing Systems Inc., the assignee of the present invention, discloses a test and alignment system for electronic display devices. The system includes a plurality of color CCD cameras mounted on a frame and arranged in an array. The rows of CCD cameras in the array are vertically adjustable and the CCD cameras in each row are laterally adjustable to allow the system to test and align different sized electronic display devices. The image output of the CCD cameras is conveyed to a computer for processing. The computer controls a test pattern generator connected to the electronic display device being tested so that the electronic display device displays appropriate test patterns of which images are captured by the CCD cameras. A wobulator is mounted on the frame and is moveable between an operable position adjacent the electronic display device and a retracted position outside of the fields of view of the CCD cameras. The wobulator is energizable to bend electron beams within the electronic display device when in the operable position. Photodiodes, also mounted on the frame, measure the luminance of the electronic display device as the wobulator bends the electron beams therein and provide input to the computer for processing.

The computer performs a comprehensive series of test and alignment functions on the electronic display device based on the image output of the CCD cameras and the output of the photodiodes as the test patterns are displayed. The test and alignment functions include color purity, focus, static and dynamic convergence, yolk rotation, orthogonality, video pattern size and centering, raster size and centering, linearity, geometry and luminance measurements and are all performed at a single station. During the series of test and alignment functions, the computer displays graphical indicators which are used by an operator to determine if the electronic display device passes or fails a test.

U.K. patent application No. 2,255,700 published on Nov. 11, 1992 and assigned to Samsung Electron Devices Company Ltd. discloses a system for measuring cathode ray tube (CRT) characteristics. The system comprises a camera, a selector, a video processor, a monitor, a magnetic field controller, a CPU and an output unit. The camera includes an array of CCD cameras as well as a plurality of magnetic field coils which are energized by the magnetic field controller under control of the CPU to move an electron beam within the CRT. Image output of the CCD cameras is conveyed to the CPU for processing. In one embodiment, the CCD cameras are movably mounted on a support to allow the position of the CCD cameras relative to the CRT under examination to be adjusted.

Although the above-described references show test and alignment systems for electronic display devices including a plurality of CCD cameras which are moveable relative to the electronic display device under examination and allow at least one characteristic of the electronic display device to be measured, improved systems for testing and aligning electronic display devices are continually being sought.

It is therefore an object of the present invention to provide a novel test and alignment system for an electronic display device and a test fixture for the same.

SUMMARY OF THE INVENTION

According to one aspect of the present invention there is provided a test fixture for a test and alignment system for an electronic display device comprising:

a frame;

a plurality of close-up optical sensors mounted on said frame to sense and produce image signals corresponding to small areas of an image displayed on an electronic display device positioned adjacent said test fixture; and

a plurality of wide-angle optical sensors mounted on said frame behind said close-up optical sensors, said wide-angle optical sensors for sensing and producing image signals corresponding to large areas of said image displayed on said electronic display device.

Preferably, the wide-angle optical sensors have fields of view which include the small areas of the displayed image and which overlap so that the wide-angle optical sensors sense and produce image signals corresponding substantially to the entire image displayed on the electronic display device. It is also preferred that the close-up optical sensors are positioned on the frame so as to be outside of the overlapping fields of view of the wide-angle optical sensors.

In one embodiment, the close-up optical sensors are arranged in an array with the rows of close-up optical sensors in the array being mounted on vertically adjustable, angled rails. The optical sensors in each row are also moveable laterally along the rails to allow the position of the optical sensors to be adjusted to accommodate different sized electronic display devices. Preferably, the array includes a centrally positioned, vertically adjustable close-up optical sensor mounted on a cantilever extending from the frame.

It is also preferred that the test fixture includes at least one wobulator mounted on the frame for creating a magnetic field to bend electron beams within the electronic display device. In one embodiment, the wobulator includes at least one pair of spaced coils mounted on the frame which are energizable to create an alternating magnetic field sufficient to encompass generally the entire electronic display device. It is however preferred that the wobulator includes a pair of vertically spaced, generally horizontal coils and a pair of laterally spaced, generally vertical coils with each of the pairs of coils being energizable to create the alternating magnetic fields and being fixed to the frame in positions outside of the field of view of the close-up and wide-angle optical sensors.

According to another aspect of the present invention there is provided a test and alignment system for an electronic display device comprising:

- a test pattern generator to be connected to an electronic display device for causing said electronic display device to generate images of video test patterns;
- a test fixture including a frame supporting a plurality of close-up optical sensors to sense and produce image signals corresponding to small areas of images displayed on said electronic display device and a plurality of wide-angle optical sensors behind said close-up optical sensors for sensing and producing image signals corresponding to large areas of images displayed on said electronic display device;
- a computer controlling said test pattern generator and processing said image signals to perform a series of tests on said electronic display device; and
- a display to provide a visual indication of the results of said series of tests.

According to still yet another aspect of the present invention there is provided a test fixture for a test and alignment system for an electronic display device comprising:

- a frame;
- at least one optical sensor mounted on said frame to sense and produce image signals corresponding to an image displayed on an electronic display device positioned adjacent said test fixture; and
- at least one wobulator mounted on said frame and including at least one pair of spaced coils energizable to create an alternating magnetic field to bend electron beams within said electronic display device, said coils being fixed to said frame outside of the field of view of said at least one optical sensor.

The present invention provides advantages in that the close-up optical sensors in combination with the wide-angle optical sensors mounted on the frame allow small areas and large areas of images displayed on the electronic display device to be captured simultaneously without requiring the close-up optical sensors to be moved. This allows a comprehensive series of test and alignment functions to be performed quickly on the electronic display device. Also, the wobulator design allows the coils to be permanently mounted on the frame outside of the fields of view of the close-up and wide-angle optical sensors while still allowing it to generate the desired alternating magnetic fields to bend electron beams within the electronic display device. This further speeds up the test and alignment procedure.

BRIEF DESCRIPTION OF THE DRAWINGS

An embodiment of the present invention will now be described more fully with reference to the accompanying drawings in which:

FIG. 1 is a block diagram of a test and alignment system for an electronic display device in accordance with the present invention;

FIG. 2*a* is a perspective view of a test fixture forming part of the test and alignment system illustrated in FIG. 1;

FIG. 2*b* is a top cross-sectional view of the test fixture illustrated in FIG. 2*a*;

FIG. 2*c* is a side cross-sectional view of the test fixture illustrated in FIG. 2*a*;

FIG. 3 is an enlarged cross-sectional view of a portion of the test fixture illustrated in FIG. 2*a*;

FIG. 4 is an enlarged cross-sectional view of another portion of the test fixture illustrated in FIG. 2*a*;

FIGS. 5 to 10 are diagrams illustrating graphical panels displayed to an operator during operation of the test and alignment system illustrated in FIG. 1; and

FIG. 11 is a front elevational view of a calibration grid used to calibrate wide-angle cameras forming part of the test and alignment system illustrated in FIG. 1.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to FIG. 1, a schematic illustration of a test and alignment system for an electronic display device 12 including CRT assemblies and television sets or computer monitors is shown and is generally indicated to by reference numeral 10. System 10 includes a test fixture 14 on which are mounted a plurality of optical sensors 16 to capture images of test patterns displayed on the electronic display device 12 and to generate video image signals corresponding to the captured images. A wobulator 18 is also mounted on the test fixture 14 to bend electron beams within the electronic display device 12 as will be described.

The video image signal output of the optical sensors 16 is conveyed to a computer 20 by way of a multiplexer and high speed video input channel processor generally indicated to by reference numeral 22. The computer 20 is connected to an operator display 24, to a test pattern generator 26 and to the wobulator 18. The test pattern generator 26 feeds test patterns to the electronic display device 12 for display under the control of the computer 20. The computer 20 can also be connected to a programmable logic controller (PLC) 28 for controlling movement of a conventional conveyor belt 30, on which electronic display devices are supported, to position each electronic display device to be tested in front of the test fixture 14. A power supply including deflection amplifiers and video amplifiers, heater and high voltage power sources, commonly referred to as a scan unit 32, that is required to operate the electronic display device 12 is also shown for reference.

The computer 20 analyses the video image signal output of the optical sensors 16 after being processed by the multiplexer and video input channel processor 22 and performs a comprehensive series of tests and alignment functions to determine the condition of the electronic display device 12. The test and alignment functions performed by the computer 20 include color purity, focus, static and dynamic convergence, yoke rotation, perpendicularity or orthogonality, video pattern size and centering, raster size and centering, linearity, geometry, and luminance measurements including brightness cut-off. During testing, the computer 20 generates graphical indicators which are displayed on the operator display 24 to allow the operator to determine easily if the electronic display device 12 passes or fails each test.

Referring now to FIGS. 2a to 2c, the test fixture 14 is better illustrated. As can be seen, the test fixture 14 includes a box-like frame 50 which accommodates the optical sensors 16. The optical sensors 16 are arranged in two sets and include an array 52 of close-up digital color CCD cameras 54 and an array 56 of wide-angle digital color CCD cameras 58. In this particular example shown, array 52 includes three rows with each row having three CCD cameras 54 and array 56 includes two rows with each row having two CCD cameras 58. Those of skill in the art will however appreciate that the number of CCD cameras shown is for illustrative purposes only and that the test fixture 14 may include a significantly larger number of CCD cameras 54 and 58 arranged in larger arrays 52 and 56.

The CCD cameras 54 in the upper and lower rows of array 52 are supported by carriages 60 mounted on generally horizontal rails 62. The rails 62 extend between a pair of generally vertical rails 64 fixed to the frame 50. The ends of the rails 62 are received by drives 66 in the form of servo-motors. The drives 66 are movable along the vertical rails 64 in response to input from a motor controller 68 (see FIG. 1), operated by the computer 20 to adjust the vertical position of the upper and lower rows of CCD cameras 54. The upper surface 62a of the upper rail 62 is downwardly inclined and the upper surface 62a of the lower rail is upwardly inclined so that the CCD cameras 54 supported by the rails are angled towards the electronic display device 12.

The outer CCD cameras 54 of the middle row of array 52 are supported by carriages 60 mounted on stubs 70. Each stub 70 is held by a drive 66 moveable along a respective rail 64 in response to motor controller 68 to adjust the vertical position of these CCD cameras 54. The stubs 70 extend inwardly and rearwardly from the drives 66 so that the CCD cameras 54 supported by the stubs are also angled towards the electronic display device 12. The central CCD camera 54 of the middle row of array 52 is supported by a carriage 72 mounted on a cantilever 74 by way of a hinge 75. Cantilever 74 extends from a face plate 76 secured to the back of frame 50.

Each of the carriages 60 accommodates a drive 78 which is responsive to motor controller 68 to move the carriage 60 along the rail 62 or stub 70 and therefore, adjust the lateral position of the CCD camera 54 supported by the carriage. As can be seen in FIG. 3, each drive 78 includes a motor 80 for rotating a shaft 82 extending into a channel 84 in the rail 62 or stub 70 by way of a transmission 85. A pinion 86 is mounted on the shaft 82 and is positioned within the channel 84. The pinion engages a rack 88 formed on a wall of the rail 62 or stub 70. Guides 90 and 92 extending from the carriages 60 are accommodated by complimentary grooves 94 and 96 formed in the rails 62 and stubs 70 to guide the carriages 60 as they move laterally along the rails and stubs.

Unlike carriages 60, carriage 72 is not laterally adjustable. Carriage 72 however supports a drive 100 responsive to motor controller 68 to pivot the carriage relative to the cantilever 74 by way of hinge 75 and adjust its position and hence the position of the central CCD camera 54. Drive 100 is best shown in FIG. 4 and includes a motor 92 and a transmission 93 for rotating a lead screw 102 cooperating with a stationary nut 104 mounted on the cantilever 74.

Referring back to FIG. 3, one of the CCD cameras 54 is better illustrated. As can be seen, the CCD camera is equipped with a lens 110, such as a Tamron 50 MM lens, sold under model number 402026 by Tamron Corp. of Japan. A CCD chip 112 is disposed behind the lens 110 at a desired distance so that a certain small detail of the entire image displayed on the electronic display device 12 is in focus when it strikes the CCD chip. The CCD chip 112 is mounted on a sliding assembly 114, which is able to reciprocate in a linear direction parallel to the optical axis 116 through lens 110. The sliding assembly 114 is abutted against and guided by a linear slide element 118, and is driven by a motor 120 via a lead-screw assembly 122. The motor 120 is responsive to motor controller 68 to position the CCD chip 112 in relation to the lens 110 so that the image displayed on the electronic display device 12 and projected onto CCD chip 112 by lens 110 is sharp and in focus.

Referring again to FIGS. 2a to 2c, mounted on the face plate 76 and surrounding the cantilever 74 is the array 56 of wide-angle CCD cameras 58. The wide-angle CCD cameras 58 have overlapping fields of view so that the video image signal output of the wide-angle CCD cameras 58 represents basically the entire displayed image on the electronic display device 12.

The inclined orientation of the CCD cameras 54 in the upper and lower rows of array 52 and the positions of the CCD cameras 54 supported on the stubs 70 and cantilever 74 result in the CCD cameras 54 being outside of the fields of view of the wide-angle CCD cameras 58. Thus, even though the wide-angle CCD cameras 58 are mounted on the frame 50 behind the close-up CCD cameras 54, the wide-angle CCD cameras 58 have unobstructed fields of view. As such, images can be captured simultaneously by both arrays of CCD cameras. Furthermore, it has been found that the inclined orientation of the CCD cameras 54 improves resolution as compared with in-line oriented CCD cameras due to the convex nature of most electronic display devices.

The wobulator 18 is mounted on the frame 50 adjacent the electronic display device 12. The wobulator 18 includes a pair of laterally spaced, generally vertical coils as shown in FIG. 2a. A pair of vertically spaced, generally horizontal coils may also be utilized in the wobulator 18 but are not illustrated. The coils are secured to the frame 50 and positioned such that they are outside of the fields of view of the CCD cameras 54 and 58. The coils are wound and dimensioned to create symmetrical alternating magnetic fields, when energized, that are sufficient to encompass substantially the electronic display device 12 under test and bend electron beams within the electronic display device.

The computer 20 includes a number of software routines which are executed during operation of the test and alignment system 10. In particular, the computer 20 includes a set-up routine which includes CCD camera 54 position information for the various sized electronic display devices which are to be tested by the test and alignment system 10. When the computer 20 executes the set-up routine, the computer supplies output to the motor controller 68 which in turn provides input to the drives 66 and 78 to move the rails 62, stubs 70 and/or carriages 60 and 72 to position the CCD cameras 54 at the proper desired positions. The computer 20 also includes conventional Image Analysis Software to determine if the CCD cameras 54 and 58 are in focus. If the CCD cameras are not in focus, the computer 20 provides output to the motor controller 68 which in turn operates motors 120 to adjust the positions of the CCD chips 112 by way of lead-screw assemblies 114. The computer also includes a test and alignment routine which when executed by the computer 20 allows the video image signal output of the CCD cameras 54 and 58 that is received by the computer to be processed and a series of tests to be performed on the electronic display device. The series of tests performed by test and alignment system 10 is similar to the series of tests disclosed in PCT application serial No. PCT/CA95/00352

filed on Jun. 13, 1995 designating the United States, the contents of which are incorporated herein by reference.

Calibration of the Wide-Angle Cameras

The fields of view of the wide-angle color CCD cameras 58 of array 56 must be calibrated to identify and store for reference, the relationship of camera pixel locations to the corresponding points in their respective fields of view, and further to ensure that their fields of view overlap and encompass the desired area of interest. The preferred method to achieve the above will now be described.

Referring now to FIG. 11, a calibration grid 130 is shown in negative form for ease of illustration. Grid 130 is in the form of a sheet having a plurality of rows of symmetrically geometrically shaped, for example rectangular, or in the preferred form, circular, apertures 131 spaced equally in each axis therethrough. The apertures 131 are formed in the sheet with precision, so that the distances between the centers of the adjacent apertures are known to a high degree of accuracy. A preferred method of creating grid 130 is by computer aided design (CAD) methods and further by photoplotting the design on high stability photographic film. For the calibration process, suitable back-lighting is applied. Although the grid 130 is illustrated in negative form, those of skill in the art will appreciate that in reality the apertures 131 are transparent and the surrounding areas of the grid 130 are opaque.

At spaced locations on the sheet, corresponding to the spacing of the wide-angle CCD cameras 58 in the array 56, the apertures 132 are identifiably different from all other apertures in the grid 130. For example, these special apertures may have different shapes or sizes. In the preferred embodiment, they are differentiated in effect by size to the extreme, by omitting them entirely. This results in larger, generally rectangular opaque areas 132 being provided on the grid 130.

During the computer image analysis process of the calibration grid, the location of the omitted apertures 132 are calculated by interpolating the coordinates of the adjacent apertures 133. These locations of the omitted apertures 132 are regarded as the central points of the region of interest 134 for each of the wide-angle CCD cameras 58. Since the distance between adjacent apertures in each row is known, the field of view of each CCD camera 58 is determined by counting the number of apertures 131 on grid 130 within the field of view along the four orthogonal directions extending from the centers of the field of view. Along each adjacent edge of the fields of view of adjacent CCD cameras 58 is located a row 135 or a column 136 of apertures 131 that is within the regions of interest of both adjacent CCD cameras. Thus their regions of interest 134 overlap. This arrangement facilitates the accurate spatial connections of areas viewed by the separate wide-angle color CCD cameras 58 of array 56, so that they encompass the entire desired area of interest.

During the application of the system 10, the electronic display device 12 is located in the generally desired location in relation to test fixture 14, where the calibration was performed using the grid 130. The test pattern applied to the electronic display device 12 for the evaluation or alignment of the image geometry displayed thereon is constructed of generally rectangularly shaped dots such that some of the dots fall within the overlapping areas of adjacent CCD cameras 58. By correlating the locations of the dots viewed by the adjacent CCD cameras, and further applying trigonometric calculations that are known to those of skill in the art, the exact location of dots displayed by the electronic display device 12 can be identified in all three dimensions. Furthermore, by including the geometric constants relating to the electronic display device 12, such as the radius of curvature and the thickness and refraction constant of glass utilized, corrections can be made for all major error sources by computer calculations. In effect, the CCD cameras 58 of array 56 can analyze the displayed image as if it were viewed from an infinite distance, which is advantageous for the definition of geometric errors.

Test and Alignment System Operation

The operation of the test and alignment system will now be described. Initially, the computer 20 executes the set-up routine to position the CCD cameras 54 at the proper positions so that the fields of view of the CCD cameras are directed towards the electronic display device 12 to be tested and aligned. Following that, the computer 20 conditions the test pattern generator 26 to feed test patterns to the electronic display device. The electronic display device 12 in turn displays the test patterns. The CCD cameras 54 and 58 which are maintained in focus, capture images of the displayed test patterns. The close-up CCD cameras 54 focus in on small details of the displayed image and therefore, capture small areas of the entire test pattern displayed on the electronic display device 12. On the other hand, the wide-angle CCD cameras 58, which have overlapping fields of view, collectively capture an image of the entire test pattern displayed on the electronic display device.

The video image signals output by the CCD cameras 54 and 58 are multiplexed and processed by the multiplexer and high speed video input channel processor 22. The high speed video input channel processor 22 performs a variety of fundamental image processing operations on the video image signals including digitizing after multiplexing, integration, thresholding, interpolation, histogram analysis as well as horizontal and vertical projections. The video input channel processor also converts the video image signals from luminance and chrominance values to red, green and blue values for processing by the computer 20.

As mentioned previously, the computer 20 analyses the video image signal output of the CCD cameras 54 and 58 to perform a comprehensive series of tests on the electronic display device when executing the test and alignment routine. The modular nature of the test and alignment routine allows an operator to condition the computer 20 to perform the tests in virtually any order and re-perform one or more of the tests if desired. The various tests which the computer 20 can perform will now be described with reference to FIGS. 5 to 10.

Focus

To measure focus, the video image signals output by the close-up CCD cameras 54 are analyzed by computer 20. A technique in software is used involving first measuring line width for the spot size and then measuring the depth of contrast which is a Modulation Transfer Function (MTF). Other focus determination methods based on analysis of pixel intensity histogram or dimensions of small features displayed on the electronic display device 12 such as dots, may be applied. By determining the center of the mass and measuring the spot size at the center of the depth of contrast, the dot correlates very well to the human eye's perception of good focus. During this test, the computer 20 generates a graphical panel for display on the operator display 24 together with a "PASS" or "FAIL" readout, as shown in FIG. 5. The focussing voltage applied to the electronic display

device 12 may be adjusted by the scan unit 32 to improve the focus of the electronic display device. The electronic display device under test is considered to be focused within the specified limits when the red "ball" is located within the dotted rectangle. The closer the red "ball" is to the center of the crosshairs the better the focus of the electronic display device 12.

Color Purity/Vertical Raster Shift/Yoke Rotation

During the "Color Purity/Vertical Raster Shift/Yoke Rotation" test, the computer 20 generates a graphical panel as shown in FIG. 6 for display on the operator display 24. The graphical panel includes several video meters which indicate adjustments to the yoke and two purity rings of the electronic display device. As the operator moves the yoke backward and forward, the "balls" in both of the upper, outside square-boxed meters move from the outside towards the center of the screen and back and the lower middle meter pointers move up and down. Rotation of the yoke will register in both of the lower outside "Tilt" meters. Movement of the two purity rings will register in all three of the upper meters. As well, the upper middle meter will indicate vertical raster shift related to the top to bottom position of the indicator "ball". The operator display 24 also displays a "PASS" or "FAIL" readout dependent on whether or not the adjustment specifications have been met.

During color purity and luminance measurements, the wobulator 18 is energized to bend electron beams within the electronic display device 12 as the images are being captured by the CCD cameras 54 and 58. Specifically, the vertical coils of wobulator 18 are simultaneously energized by an alternating polarity current. The coils generate a vertical symmetrical alternating magnetic field which encompasses the electronic display device and deflects horizontally, the electron beams impinging upon the phosphors in the electronic display device. The alternating symmetrical deflection of the electron beams in relation to the phosphor dots or phosphor stripes in the electronic display device result in equal reduction of emitted light intensity provided the electron beams are aligned with the center of the phosphor dots or stripes when no magnetic field is generated by the wobulator 18. Any difference in light intensity is an indication of discrepancy in beam landing.

Following this, the horizontal coils of wobulator 18 (if included) are simultaneously energized to generate a symmetrical alternating horizontal magnetic field which encompasses the electronic display device and deflects vertically, the electron beams impinging upon the phosphor dots or stripes in the electronic display device.

While the wobulator coils are being energized, the video image signals generated by the close-up CCD cameras 54 are analyzed and processed by the computer 20 to determine beam landing discrepancy. If desired, the video image signals generated by the wide-angle CCD cameras 58, or a selected Region of Interest thereof, can be analyzed and processed by the computer 20 to determine beam landing discrepancy.

Static Convergence

During the Static Convergence test, the computer 20 generates a graphical panel as shown in FIG. 7, for display on the operator display 24. The graphical panel reflects the result of adjustments to the magnetic correction device of the electronic display device, traditionally two pairs of yoke rings known as the four pole ring magnets and the six pole ring magnets. The operator adjusts the two pairs of yoke rings to bring the red and blue crosses as close to the center of the crosshairs as possible. During this test, the central CCD camera 54 is positioned at the center of the electronic display device and the video image signals generated by the CCD camera 54 are analyzed and processed by the computer 20. The static convergence of the electronic display device under test, as an example, is acceptable when the red and blue crosses are adjusted so that they are located within the dotted rectangle and no further than half of the width of the dotted rectangle apart.

Dynamic Convergence

During the Dynamic Convergence test, the computer 20 generates a graphical panel as shown in FIG. 8, for display on the operator display 24. The graphical panel illustrates convergence as typically measured at four or more locations on the electronic display device. During this test, the video image signals from a number of the CCD cameras 54 are analyzed and processed by the computer 20. These four locations in the example are along the horizontal and vertical axes of the electronic display device, near to the edge of the active display area. A combined result is displayed in the center meter. If the "ball" is within the dotted rectangle, the four convergence measurements are within the specified limits. Yoke tilt is also displayed. If both Yoke tilt and measured convergence are within specified limits, a "PASS" result is obtained.

During both convergence measurements, the individual dots of color on the electronic display device cannot be analyzed by themselves for two reasons. The color dots on an electronic display device are not perfectly round but typically have very odd shapes, and due to the large cross-section of the electron beam, a group of dots are illuminated. To compensate for irregularity in beam shape, a group of dots for each color is analyzed and the relative position of the centers of their masses are calculated mathematically. By analyzing the projection, the irregular beam shapes can be compensated for to provide a more accurate measurement.

Orthogonality

The system 10 also tests the orthogonality of the horizontal and vertical windings of the deflection yoke and displays a graphical panel on operator display 24 which includes an "X" and "Y" gauge as well as a "PASS" or "FAIL" readout, as shown in FIG. 9. The method used to determine the orthogonality of the windings is well known to those of skill in the art.

Other Geometric Features

Similar to orthogonality, a variety of other geometric tests that are well known to those of skill in the art can be performed on the electronic display device 12 by the test and alignment system 10 such as for example, pincushion and gull-wing distortion, linearity etc.

As mentioned previously, if desired, one or more of the above tests can be re-performed at any time during execution of the test and alignment routine. For example, FIG. 10 shows the graphical panel output by computer 20 for display on the operator panel 24 during re-performance of the "Color Purity/Vertical Raster Shift/Yoke Rotation" test.

As those of skill in the art will appreciate, the test and alignment system allows close-up and wide-area images of test patterns displayed on an electronic display device to be captured simultaneously permitting a series of tests to be performed on the electronic display device quickly and in real-time.

Although a preferred embodiment of the present invention has been described, those of skill in the art will appreciate that other variations and modifications may be made without departing from the scope thereof as defined by the appended claims.

We claim:

1. A test fixture for a test and alignment system for an electronic display device comprising:

a frame;

a plurality of close-up optical sensors mounted on said frame to sense and produce image signals corresponding to small areas of an image displayed on an electronic display device positioned adjacent said test fixture; and

a plurality of wide-angle optical sensors mounted on said frame behind said close-up optical sensors, said wide-angle optical sensors being calibrated using a calibration grid so as to have overlapping fields of view to sense and produce image signals corresponding to substantially the entire image displayed on said electronic display device, wherein said close-up optical sensors are positioned on said frame in front of said wide-angle optical sensors so as to be outside of the fields of view of said wide-angle optical sensors, at least some of said close-up optical sensors being moveably mounted on said frame.

2. A test fixture as defined in claim 1 wherein said close-up optical sensors include a plurality of rows of optical sensors, said optical sensors being angled in a direction towards said electronic display device.

3. A test fixture as defined in claim 2 further including at least one intermediate central row of optical sensors.

4. A test fixture as defined in claim 3 wherein said central row of optical sensors includes a vertically movable central optical sensor, said central optical sensor being positioned on said frame so that the field of view thereof includes the center of said image.

5. A test fixture as defined in claim 4 wherein said central optical sensor is pivotally mounted on said frame.

6. A test fixture as defined in claim 5 wherein said central optical sensor is pivotally mounted on a cantilever.

7. A test fixture as defined in claim 2 wherein said rows of optical sensors are supported on inclined rails mounted on said frame, said rails being vertically adjustable.

8. A test fixture as defined in claim 7 wherein the optical sensors in said rows are movable laterally along said respective inclined rails.

9. A test fixture as defined in claim 8 wherein the optical sensors in said rows include drive means to move said optical sensors laterally along said rails.

10. A test fixture as defined in claim 1 further including at least one wobulator mounted on said frame for creating an alternating magnetic field to bend electron beams within said electronic display device.

11. A test fixture as defined in claim 10 wherein said at least one wobulator includes at least one pair of laterally spaced, elongate electromagnets mounted on said frame outside of the fields of view of said close-up and wide-angle optical sensors and being energizable to generate a generally uniform alternating magnetic field, said alternating magnetic field encompassing generally the entire display screen of said electronic display device.

12. A test fixture as defined in claim 11 wherein said at least one wobulator includes a pair of vertically spaced, generally horizontal electromagnets and a pair of laterally spaced, generally vertical electromagnets, said pairs of electromagnets being energizable to generate alternating magnetic fields.

13. A test and alignment system for an electronic display device comprising:

a test pattern generator to be connected to an electronic display device for causing said electronic display device to generate images of video test patterns on said electronic display device;

a test fixture including a frame supporting a plurality of close-up optical sensors to sense and produce image signals corresponding to small areas of images displayed on said electronic display device and a plurality of wide-angle optical sensors to sense and produce image signals corresponding to larger areas of images displayed on said electronic display device, said close-up optical sensors being positioned on said frame so as to be outside of the fields of view of said wide-angle optical sensors, at least some of said close-up optical sensors being moveable relative to said frame, said wide-angle optical sensors being calibrated using a calibration grid so as to have overlapping fields of view to sense and produce image signals corresponding to substantially the entire image displayed on said electronic display device;

a computer controlling said test pattern generator and processing said image signals to perform a series of tests on said electronic display device; and

a display to provide a visual indication of the results of said series of tests.

14. A test and alignment system as defined in claim 13 wherein said close-up optical sensors include a plurality of rows of optical sensors, said optical sensors being angled in a direction towards said electronic display device.

15. A test and alignment system as defined in claim 14 further including at least one intermediate central row of optical sensors.

16. A test and alignment system as defined in claim 15 wherein said central row of optical sensors includes a vertically movable central optical sensor, said central optical sensor being positioned on said frame so that the field of view thereof includes the center of said image.

17. A test and alignment system as defined in claim 16 wherein said central optical sensor is pivotally mounted on said frame.

18. A test and alignment system as defined in claim 17 wherein said central optical sensor is pivotally mounted on a cantilever.

19. A test and alignment system as defined in claim 14 wherein said rows of optical sensors are supported on inclined rails mounted on said frame, said rails being vertically adjustable.

20. A test and alignment system as defined in claim 19 wherein the optical sensors in said rows are movable laterally along said respective inclined rails.

21. A test and alignment system as defined in claim 20 wherein the optical sensors in said rows include drive means to move said optical sensors laterally along said rails.

22. A test and alignment system as defined in claim 13 further including at least one wobulator mounted on said frame for creating an alternating magnetic field to bend electron beams within said electronic display device.

23. A test and alignment system as defined in claim 22 wherein said at least one wobulator includes at least one pair of laterally spaced electromagnets mounted on said frame and being energizable to generate a generally uniform alternating magnetic field, said alternating magnetic field encompassing generally the entire display screen of said electronic display device.

24. A test and alignment system as defined in claim 23 wherein said at least one wobulator includes a pair of

vertically spaced, generally horizontal electromagnets and a pair of laterally spaced, generally vertical electromagnets, said pairs of electromagnets being energizable to generate alternating magnetic fields.

25. A test fixture for a test and alignment system for an electronic display device comprising:

a frame;

at least one optical sensor mounted on said frame to sense and produce image signals corresponding to an image displayed on an electronic display device positioned adjacent said test fixture; and

at least one wobulator mounted on said frame and including at least one pair of laterally spaced electromagnets, said electromagnets being energizable to generate a generally uniform alternating magnetic field generally encompassing the entire display screen of said electronic display device to bend electron beams within said electronic display device, said electromagnets being fixed to said frame outside of the field of view of said at least one optical sensor.

26. A test fixture as defined in claim 25 wherein said at least one wobulator includes a pair of vertically spaced, generally horizontal electromagnets and a pair of laterally spaced, generally vertical electromagnets, said pairs of electromagnets being energizable to generate alternating magnetic fields.

27. A test fixture as defined in claim 1 wherein at least one of said wide-angle and close-up optical sensors includes a CCD device and a fixed lens element, said CCD device being movable relative to said lens element to change the focus of said at least one optical sensor.

28. A test fixture as defined in claim 27 wherein said CCD device is mounted on a slide assembly responsive to a drive, said slide assembly being slidable along an axis parallel to an optical axis extending through said at least one optical sensor.

29. A test and alignment system as defined in claim 13 wherein at least one of said wide-angle and close-up optical sensors includes a CCD device and a fixed lens element, said CCD device being movable relative to said lens element to change the focus of said at least one optical sensor.

30. A test and alignment system as defined in claim 29 wherein said CCD device is mounted on a slide assembly responsive to a drive, said slide assembly being slidable along an axis parallel to an optical axis extending through said at least one optical sensor.

31. A test fixture for a test and alignment system for an electronic display device comprising:

a frame;

a plurality of close-up cameras mounted on said frame to sense and produce image signals corresponding to small areas of an image displayed on an electronic display device positioned adjacent said test fixture; and

a plurality of wide-angle cameras mounted on said frame behind said close-up cameras and being calibrated using a calibration grid so as to have overlapping fields of view, said wide-angle cameras sensing and producing image signals corresponding to substantially the entire image displayed on said electronic display device, said close-up cameras being positioned on said frame in front of said wide-angle cameras outside of the fields of view of said wide-angle cameras.

32. A test fixture as defined in claim 31 wherein said close-up cameras are arranged in an array including a plurality of vertically spaced rows, each of said rows including a plurality of close-up cameras, said close-up cameras being angled in a direction towards said electronic display device to compensate for convexities in the shape of said electronic display device.

33. A test fixture as defined in claim 32 wherein said rows of close-up cameras are supported on inclined rails mounted on said frame, said rails being vertically adjustable.

34. A test fixture as defined in claim 33 wherein said close-up cameras are movable along said respective inclined rails.

35. A test fixture as defined in claim 31 wherein each of said wide-angle and close-up cameras includes a CCD device and a fixed lens element, said CCD device being moveable relative to said lens element to change the focus of said camera.

36. A test fixture as defined in claim 31 wherein said calibration grid includes an identifier for each of said wide-angle cameras, each identifier defining a central point for the field of view of said respective wide-angle camera.

37. A test fixture as defined in claim 36 wherein said calibration grid includes an array of apertures formed in a sheet, each said identifier being constituted by a region of said sheet devoid of apertures.

* * * * *